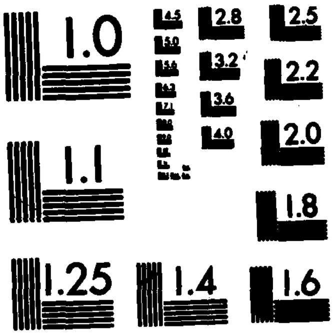
AD-A125 070 CEDA HIGH-LEVEL SOFTWARE SYSTEM DESIGN(U) MITRE CORP 1/2
BEDFORD MA J R CALABRO ET AL. JAN 83 MTR-8651
ESD-TR-82-419 F19628-82-C-0001

UNCLASSIFIED F/G 9/2 NL


1.0
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

(12)

ESD-TR-82-419 MTR-8651

CEDA HIGH-LEVEL SOFTWARE SYSTEM DESIGN

By
J. R. CALABRO
M. M. POZZO

JANUARY 1983

Prepared for
DEPUTY FOR DEVELOPMENT PLANS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
Hanscom Air Force Base, Massachusetts

Approved for public release;
distribution unlimited.

Project No. 496B
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract No. F19628-82-C-0001

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved for publication.

Edwin W Knox

EDWIN W. KNOX, 2Lt USAF
Project Officer, C2 CONCAP

FOR THE COMMANDER

Charles J LaBlonde

CHARLES J. LaBLONDE, Lt Col, USAF
Acting Director
Advanced Concepts and Technology
Deputy for Development Plans

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ESD-TR-82-419	2. GOVT ACCESSION NO. AD-A125 070	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) CEDA HIGH-LEVEL SOFTWARE SYSTEM DESIGN		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER MTR-8651
7. AUTHOR(s) J.R. CALABRO M.M. POZZO		8. CONTRACT OR GRANT NUMBER(s) F19628-82-C-001
9. PERFORMING ORGANIZATION NAME AND ADDRESS The MITRE Corporation P.O. Box 208 Bedford, MA 01730		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Project No. 496B
11. CONTROLLING OFFICE NAME AND ADDRESS Deputy for Development Plans Electronic Systems Division, AFSC Hanscom Air Force Base, MA 01731		12. REPORT DATE January 1983
		13. NUMBER OF PAGES 107
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

DECISION AIDS	MODULAR DESIGN
HUMAN FACTORS	MODULAR APPROVAL
AUTOMATED TACTICAL AIDS	Command and Control
DATA RECORDING	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Air Force is involved in research in the area of human performance and training; specifically the interaction of human factors with the effectiveness of C^2 systems is being studied. A software system which is a Capability for Evaluating Decision Aids (CEDA) was developed using the ESD CONCAP capability at MITRE. CEDA will be used by the Human Resources Laboratory for testing human performance on a tactical operations problem as the amount of information and form of presentation are varied.

(over)

The paper is a high level description of the software design. It is assumed that the reader knows computer programming and the terms common to DEC's RSX-11M Operating System.

ACKNOWLEDGMENTS

This report has been prepared by The MITRE Corporation under Project No. 496B. The contract is sponsored by the Electronic Systems Division, Air Force Systems Command, Hanscom Air Force Base, Massachusetts.

The authors would like to thank Judy Clapp, Bob Hanckel and Dick Wagner for their help in putting this manual together and Joan Delmore for her secretarial and editing assistance.

TABLE OF CONTENTS

Section Page

TABLE OF CONTENTS (Concluded)

LIST OF ILLUSTRATIONS

SECTION 1

INTRODUCTION

1.1 BACKGROUND/PURPOSE

This document presents the high-level software design of the Capability for Evaluating Decision Aids (CEDA) system, Version 1, being developed under Project 4960 by MITRE. The software was developed and is available in the ESD C2 CONCAP PDP11 Facility at MITRE. In a current application, CEDA will be used by the Human Resources Laboratory (HRL) at Wright Patterson Air Force Base as an experimental capability for testing human performance on a tactical operations problem as the amount and presentation of computer aids are experimentally varied.

This paper serves neither as a user's manual nor a maintenance manual for CEDA; a separate user's manual has been written. This, rather, is a description of the overall flow of processing and it is assumed that the reader has a general knowledge of computer systems, computer programming, and terms common to DEC's RSX-11M Operating System.

1.2 CEDA PROJECT 4960

The Human Resources Laboratory (HRL) at Wright Patterson Air Force Base is conducting in-house research on the interaction of human factors with C^2 systems. This work is being done to complement a larger effort by the Air Force in the area of human performance and training. MITRE has been tasked, under Project 4960, to put together a generic Capability for Evaluating Decision Aids which will permit analysis of operator performance as a result of automated tactical decision aids. This capability can be used to evaluate new concepts for command and control systems and to refine the human interface to automated decision aids.

To develop such a capability, the decision aids and overall capabilities of the Tactical Operations Planner (TOP) Phase 2 system were used as a model. Stimulus control features, data recording for protocol analysis, collection of subject response times, and the "pulsing" of peripheral devices for additional dependent measurements are additional features incorporated into CEDA which provide the framework whereby experiments may be defined and the results analyzed.

1.3 ORGANIZATION OF MANUAL

This manual is divided into six sections. This first section serves as an introduction. Section 2, "CEDA Functions", describes in general terms both the capabilities CEDA provides to a decision maker and the kinds of data collected from each experimental session.

Section 3, "Software Description," provides a description of the CEDA software and presents a high-level view of how CEDA operates internally. Section 4, "Files Description", describes the disk-resident files which support CEDA. Section 5, "System Flow by Module," contains a set of high-level flow diagrams of each executive task and brief descriptions of all routines to illustrate the general flow of processing within CEDA. The final section, Section 6, "Hardware System Used for Development", mentions the facility at MITRE on which CEDA was developed (C^2 CONCAP) and the system that will be used at HRL to perform the human factors study.

Appendices A, B and C include information on data collected from a session. Appendix D provides details on the Data Manipulation Language (DML) used by CEDA to retrieve and update the data base. Appendix E contains the data dictionary (record layouts) for all the files. A glossary is included at the end of this manual with terms specific to tactical operations.

SECTION 2

CEDA FUNCTIONS

2.1 USER INTERFACE

As previously stated, CEDA can be used to assist researchers in determining how the availability and design of automated decision aids affect C^2 decision making.

For the initial version of CEDA, the person responsible for session definition will be able to define the session's characteristics (i.e., function availability, data recording units) by editing an appropriate data file (see User's Manual for definition of INPUT.DAT). To change any other files used by CEDA, the user should be extremely cautious since this type of maintenance/update of CEDA is not covered in either this document or the User's Manual. It is left as the responsibility of the user to maintain and verify any changes to these files.

The initial system supports a two-terminal configuration (station): a video screen with keyboard for text input and user request for decision aids, a graphics display and a means for moving a graphics cursor, for graphics-based user input. Provisions have been made in the design for future expansion to a system which can support up to eight stations. This capability has not yet been implemented, however.

To begin a session, the user will first be prompted to complete a login process. He will then be given a number of (computer-) automated aids to assist him in solving the tactical operations problem which is presented to him. These automated aids are: graphical displays of the tactical battle management scenario, menus for requesting a variety of information and computer actions, tabular presentation of information, and computer-aided recommendations. The user will work his way through the tactical problem by invoking the various aids available to him and assessing computed recommendations.

2.2 DATA COLLECTION

One of the key experimental features of CEDA is the data collection and recording capability, which will permit a human factors analyst to correlate stimuli with user response. Throughout each user session, data is collected on the following:

- o system events - system stimuli (e.g., graphics display)
- o user responses - action taken by user after system event
- o user response times - time between system event and user response

This data is recorded on a disk file under the user´s login name (up to eight characters). The first record in the user´s file is the login record and it has the following form:

Col.	Data
1-5	´LOGIN´
7-14	user´s name
16-19	user´s rank
31-36	session date (mmddyy)
46-53	session time (hh:mm:ss)

All remaining records are either system event records or menu entry records. System event refers to any computer driven stimulus presented to the user (e.g. graphics display, menu display). Each system event is recorded as a system event record. If the event is the display of a menu, subsequent records (menu entries) are added to identify user entries made in the menu.

The format for each system event record is as follows:

Col.	Data
1-5	´EVENT´
7-10	system event code
12-21	subject response time
23-26	subject response code
28-37	if menu, time to complete menu entries otherwise, 0.0 is recorded

As stated above, if the system event is a menu, a list of user entries is also recorded, where lower case letters represent input and upper case letters represent the unaltered, menu (default) values.

Appendix A contains a list of all possible system event codes and their definitions. Appendix B contains a list of all possible user response codes and their definitions. See Appendix C for a typical session´s data collection file.

Although data recording was developed as a central part of the CEDA software, it can be turned off without affecting any of the other features.

2.3 PERIPHERAL DEVICES

In addition to stimulus/response data recording, CEDA can also interface with peripheral devices (e.g. occulometer, electroencephalograph) which can be used to measure eye movement, brain waves, etc. To do this, each system event will initiate the following sequence of pulses:

Generation of system event - pulse channel 1 (on)

Presentation of system event for:

graphics display - pulse channel 2 (on-off)
menu display - pulse channel 3 (on-off)
table display - pulse channel 4 (on-off)

User response - pulse channel 1 (off)

The pulses are internally generated in the PDP-11 minicomputer and are transmitted to the peripheral devices via a DEC DRS11-A 48 bit output module. The generation of these pulses is initiated by a CEDA software routine (SWITCH.FTN, see GENLB.OLB).

As with the data recording capability, pulsing of peripheral devices is not essential to the functioning of the CEDA software. That is, CEDA performs the same functions with or without these devices connected.

2.4 EXPERIMENTAL CONTROL FEATURES

Several other features were included in CEDA to permit well defined experiments to be performed and to control in an exact manner the user stimuli. This is needed to present analysts with reasonable data from which observations can be made.

In order to precisely measure user stimulus/response, the painting of video and graphics displays is invisible to the user. This was accomplished by installing a hardware switch in the terminals at HRL. This switch turns the screen off at the beginning of a system event (stimulus) and turns it on when the screen is complete.

The person responsible fo: session definition also has control over the number of available aids and actual location of the function key on the terminal which invokes each function. As stated above, session definition (INPUT.DAT) is more fully explained in the CEDA User's Manual.

SECTION 3

SOFTWARE DESCRIPTION

3.1 GENERAL

This section presents a general description of the software that defines CEDA. The system is broken into functional modules and each module is described individually in terms of its function and interfaces. In terms of this document, a module is a task, a separately compiled set of code that contains routines for performing a particular type of operation (e.g., table display, graphic operation, data manipulation) and any associated data files. A module is activated by a controlling module (User Function) when a request is received to perform the operations specific to that module.

Off-the-shelf software accessed by CEDA is also mentioned. Again, this manual is not to be used as a maintenance manual since it does not cover details necessary to make modifications to the CEDA software or data files. This manual only provides a discussion of the high level framework of the software system.

Entirely written in Fortran IV Plus, CEDA consists of approximately 240 routines and runs under DEC's RSX 11-M operating system. Two off-the-shelf packages are used by CEDA: DEC's Record Management Services (RMS) for data base handling and DEC's Forms Management System (FMS) for forms (menus and tables) generation. To use either of these packages or RSX 11-M, a license from DEC is required.

3.2 MODULAR DESIGN

Since CEDA, Version 1, was developed as an initial capability upon which enhancements will be added in the future, the software design is extremely modular. The total capability of CEDA was broken down by functional module so that each one could serve as an independent package. In order to use a module within another application, only its function and interfaces to the outside world need to be known to the designer. Its internal processing is irrelevant.

This modular approach to the design of CEDA also permits individual modules to be used by other systems which have similar needs to CEDA. That is, another application may be able to use the

Data Base module to access/update the CEDA data base by using the same interface (DML queries, see below). CEDA may, therefore, be considered a set of general purpose software tools which together perform all the functions needed for this particular capability. The development of general purpose software tools is a main goal of C^2 CONCAP.

Briefly, the functional modules are as follows:

System - initiates the session and closes the session for each user.

Data Recording - performs all data collection and recording.

User Function - controls all user function requests and initiates other modules which perform the processing required to satisfy each request.

Graphics - handles all graphics requests.

Menu - handles all menu requests.

Table - handles all table requests.

Data Base - handles all data base requests.

Decision Aid - handles all decision aid requests.

3.3 DATA MANIPULATION LANGUAGE

A Data Manipulation Language (DML) was developed as a means of communicating with the Data Base. A DML statement sent to the Data Base from any of the other modules can request information to be retrieved or updated. Details of the DML syntax, the available DML statements, and when to use them can be found in Appendix D.

3.4 MODULES

This section presents a description of CEDA by functional module and the interfaces for each module mentioned above. Each module is a task, a separately compiled set of routines (see Section 3.1) and any associated data files. Only three of the modules contain data files as well as tasks: the Data Base Module, the Decision Aid module and the Graphics module. The definition of these files can be found in Section 4 - FILES DESCRIPTION.

Each task described in this section has been divided into its executive (the main controlling routine for the whole module) and its supporting library (the collection of routines which is invoked by the executive and which supports the function of the module). In some cases, the library is further broken down into distinct packages of routines, that together serve a specific purpose.

For each task's executive the following is given: filename, driver (invoking task), a short description, interfaces, associated library, and approximate size in 16-bit words.

For each library the following is given: filename, driver (invoking routine), a short description, and the supporting routines.

Each supporting routine is defined by: filename, function, and interface.

Details of the calling sequence for each routine are further documented as internal comments within the specific routine. Since this manual presents the high-level system design, it assumes that any modifications to the software will require that the source code be examined first to determine feasibility and details of the modification.

SYSTEM MODULE

System Executive

Filename: SEXEC. FTN

Driven by: Task which initiates the session.

Description: This task is responsible for initiating and ending CEDA for each user. Once this task has performed initialization and user login processing, the Data Recording Executive and User Function Executive are spawned and control is passed to the User Function Executive. Control is returned to the System Executive at the conclusion of each user's session.

Interface: Input - Resident Common Block (CEDACM)
Output/Result - CEDA initialization and login information

Associated Libraries: SELB.OLB - System Executive Library
GENLB.OLB - General Library

Approximate Size: 542 words

System Executive Library

Filename: SELB.OLB

Driven by: SEXEC.FTN

Description: This library supports the function of the System Executive (SEXEC.FTN).

Supporting Routines:

Filename	Function	Input	Output/Result
SINIT.FTN	perform session initialization	resident common information	session initialization information
LEARN.FTN	define video terminal function keys and set up graphics terminal	terminal identifications and resident common information	defined function keys and and set graphics terminal
LOGIN.FTN	perform login processing	terminal identification	login information
SCLEAN.FTN	perform subject-specific clean-up	terminal identifications	closed files and detached terminals
ATTACH.FTN	attach terminal	terminal and logical unit number	attached terminal
DETACH.FTN	detach terminal	terminal and logical unit number	detached terminal

Approximate Size: 2816 words

DATA RECORDING MODULE

Data Recording Executive

Filename: RECORD.FTN

Driven by: UFEXEC.FTN, MEXEC.FTN, TEXEC.FTN, GEXEC.FTN

Description: This module is initiated at the beginning of the session and records system events, subject responses and response times. It is controlled by event flags to indicate relevant session events.

Interface: Event flags 65, 66, 67, 68, 69

Associated Library: None

Approximate Size: 511 words

USER FUNCTION MODULE

User Function Executive

Filename: UFEXEC.FTN

Driven by: SEXEC.FTN

Description: This task acts as the main interface between the user function requests and CEDA. All function requests are handled by the User Function Executive and passed to the appropriate tasks to complete processing.

Interface: Input - user function request
Output/Result - task requests

Associated Libraries - UFELB.OLB - User Function Executive Library
GENLB.OLB - General Library

Approximate Size: 876 words

User Function Executive Library

Filename: UFELB.OLB

Driven by: UFEXEC.FTN

Description: This library supports the function of the User Function Executive (UFEXEC.FTN).

Supporting Routines:

Filename	Function	Input	Output/Result
DCODE.FTN	convert function key input to function request	function key code	function request
FINTRP.FTN	convert function request into task requests	function request	task requests

Approximate Size: 1024 words

GRAPHICS MODULE

Graphics Executive

Filename: GEXEC.FTN

Driven by: UFEXEC.FTN

Description: This task is responsible for all graphics requests. As each request is received and interpreted, one of the CEDA set of graphics functions is initiated.

Interface: Input - graphics request
Output/Result - graphics function (e.g. display)

Associated Libraries: GELB.OLB - Graphics Executive Library
PLOTLB.OLB - Plot Library (graphics-terminal specific calls)
GENLB.OLB - General Library

Approximate Size: 1614 words

Graphics Executive Library

Filename: GELB.OLB

Driven by: GEXEC.FTN

Description: This library supports the function of the Graphics Executive (GEXEC.FTN) and provides an interface between CEDA and the terminal-specific graphics package used.

Supporting Routines:

Filename	Function	Input	Output/Result
GINTRP.FTN	interpret graphics request	graphics request code	graphics function request
DGRID.FTN	perform graphics grid function	grid request	displayed grid
GRID.FTN	draw a grid	grid description and color	displayed grid
ZZOOM.FTN	perform graphics zoom function	zoom level	new displayed area at new zoom level
AAREA.FTN	determine which area user has selected by cursor position	area definitions	area identification
RDCRSR.FTN	read cursor location	cursor input	cursor position
QUAD.FTN	determine in which quadrant the cursor lies	quadrant definitions	quadrant identification
DMAP.FTN	draw a map for the selected area	map file to be displayed	displayed map and foreground information

Filename	Function	Input	Output/Result
SCALE.FTN	add nautical miles scale to map	zoom level	nautical miles scale
HDCHAR.FTN	display a string of (hardware-generated) characters	character string and location	displayed character string
TPAT.FTN	display a (hardware-generated) symbol	symbol code and color	displayed symbol
NUMBER.FTN	display a (hardware-generated) number	number and location	displayed number
IDTAR.FTN	perform graphics identify function	symbol locations	identified target
IDENT.FTN	determine which symbol user has selected by cursor position	symbol locations	symbol identification
UPID.FTN	request data base update for symbol (target) identification	symbol identification	data base request
UNID.FTN	request data base to unidentify symbol (target)	symbol identification	data base request
SYMID.FTN	request data base for symbol (non-target) identification	symbol identification	data base request
MSN.FTN	perform graphics mission pairing function	symbol locations	displayed mission pairing lines
PAIR.FTN	draw a line between airbase and target to indicate mission pairing	symbol locations and line color	displayed line

Filename	Function	Input	Output/Result
GOPT.FTN	perform graphic options update function	symbol locations and identification	displayed updated symbols
BTAR.FTN	perform graphics bold (target) function	(target) identification	bold (target)
LOCATE.FTN	perform graphics locate function	symbols locations	data base request
BLINK.FTN	bold symbol during weapon allocation	symbol location	bold symbol
RAB.FTN	request data base for(airbase) symbol information	symbol identification	data base request
RTAR.FTN	request data base for (target) symbol information	symbol identification	data base request

Approximate Size: 17408 words

Plot Library

Filename: PLOTLB.OLB

Driven by: GEXEC.FTN, GELB.OLB

Description: This library supports the function of the Graphics Executive (GEXEC.FTN) and provides the terminal-specific graphics package.

Supporting Routines:

Filename	Function	Input	Output/Result
ALL.FTN	included common	n/a	n/a
PLOTS.FTN	initialize display	n/a	initialized display
PLOT.FTN	move display reference	update reference	new display location
OFFSET.FTN	set offset values	offset	set offset values
UPDATE.FTN	increment the plotting coordinates	n/a	n/a
UPORIG.FTN	update plotting origin	n/a	n/a
NEWPEN.FTN	request a color	color code	selected color
CIRI.FTN	perform initialization for circle routine	n/a	initialization for circle routine
CIRCL.FTN	draw a circle	circle	displayed location, circle and color
TOUTST.FTN	display (hardware-generated) character string	character string	displayed character string

Filename	Function	Input	Output/Result
SCURSR.FTN	read cursor-location	cursor input	cursor location
TSYM.FTN	display (hardware-generated) symbol	symbol identification and color	displayed symbol

Approximate Size: 6656 words

MENU MODULE

Menu Executive

Filename: MEXEC.FTN

Driven by: UFEXEC.FTN

Description: This task is responsible for all menu requests. When a menu request is received and interpreted, the appropriate menu is displayed, user input is accepted and interpreted, and a resulting set of tasks is requested.

Interface: Input - menu request
Output/Result - menu generation, interpretation, and request for resultant tasks

Associated Libraries:	MELB.OLB	Menu Executive Library
	HLLFOR.OLB	Fortran interface library for DEC's Forms Management Services (FMS) Package
	FDVLIB.OLB	Form Driver Library for FMS.
	GENLB.OLB	General Library
	MNLB.FLB	Library of menus for FMS

Approximate Size: 1536 words

Menu Executive Library

Filename: MELB.OLB

Driven by: MEXEC.OLB

Description: This library supports the function of the Menu Executive (MEXEC.FTN) and provides an interface between CEDA and the terminal-specific forms generation package used.

Supporting Routines:

Filename	Function	Input	Output/Result
MNCP.FTN	menu clear and paint	menu name	displayed menu
MSTART.FTN	reads all field values after user input	menu name	all field values from menu including user input
STOPBL.FTN	stops the menu title from blinking	menu name	unblinking title
RCRD.FTN	record all field values from menu including user	all field values from menu	file containing all field values from menu
MNO1H.FTN	handler for menu #1	user input	data base and task requests
MNO2H.FTN	handler for menu #2	user input	data base and task requests
MNO7H.FTN	handler for menu #7	user input	data base and task requests
MNO8H.FTN	handler for menu #8	user input	data base and task requests
MNO9H.FTN	handler for menu #9	user input	data base and task requests

Filename	Function	Input	Output/Result
MN9BH.FTN	handler for menu #9B	user input	data base and task requests
MN10H.FTN	handler for menu #10	user input	data base and task requests
MN11H.FTN	handler for menu #11	user input	prioritization
PUT.FTN	controller for displaying information in a menu prior to user input	menu name	put routine for specific menu invoked
PUT08.FTN	display target and weapon choices for selection in menu #8	DBFILE containing the target and weapon records	required information displayed in menu
PUT09.FTN	display airbase choices for selection in menu #9	DBFILE containing the airbase records	required information displayed in menu
PUT9B.FTN	display the suggested mission for inspection in menu #9B	DBFILE containing the suggested mission record	required information displayed in menu
PUT10.FTN	display all planned missions for user inspection in menu #10	DBFILE containing the planned mission records	required information displayed in menu
DBALL.FTN	builds the DML statement to retrieve all identified targets for allocation purposes	n/a	n/a

Filename	Function	Input	Output/Result
DBID.FTN	builds the DML statement to retrieve specific identified targets for allocation purposes	identified target numbers input by the user	n/a
DBMNTL.FTN	builds the DML to retrieve all planned missions for display in menu #10	n/a	data base and task requests
DBTL	builds the DML statement to retrieve all identified targets for table 3-Identified Targets Summary	n/a	data base and task requests
TBLTL	builds the task list for a table request from menu 2-Data Base Search Menu	table chosen by user	task request

Approximate Size: 18176 words

TABLE MODULE

Table Executive

Filename: TEXEC.FTN

Driven by: UFEXEC.FTN

Description: This task is responsible for all table requests. When a table request is received and interpreted, the appropriate table and information is displayed.

Interface: Input - table request
Output/Result - table generation

Associated Libraries: TELB.OLB - Table Executive Library
(See Menu Executive FMS libraries)
GENLB.OLB - General Library

Approximate Size: 1536 words

Table Executive Library

Filename: TELB.OLB

Driven by: TEXEC.FTN

Description: This library supports the function of the Table Executive (TEXEC.FTN) and provides an interface between CEDA and the terminal-specific forms generation package used.

Supporting Routines:

Filename	Function	Input	Output/Result
TB01.FTN	format data for Table #1	data base information	formatted data
TB03.FTN	format data for Table #3	data base information	formatted data
TB04.FTN	format data for Table #4	data base information	formatted data
TB04A	bring up page 1 for Table #4	n/a	page 1
TB04B	bring up page 2 for Table #4	n/a	page 2
TB04C	bring up page 3 for Table #4	n/a	page 3
TB04D	bring up page 4 for Table #4	n/a	page 4
TB04E	bring up page 5 for Table #4	n/a	page 5
TB05.FTN	format data for Table #5	n/a	formatted data
TB06.FTN	format data for Table #6	data base information	formatted data

Filename	Function	Input	Output/Result
TB07.FTN	format data for Table #7	n/a	formatted data
TB08.FTN	format data for Table #8	n/a	formatted data
TB09.FTN	format data for Table #9	n/a	formatted data
TB10.FTN	format data for Table #10	n/a	formatted data
TB11.FTN	format data for Table #11	n/a	formatted data
TB12.FTN	format data for Table #12	n/a	formatted data
TB13.FTN	format data for Table #13	n/a	formatted data
TB14.FTN	format data for Table #14	n/a	formatted data
TB15.FTN	format data for Table #15	n/a	formatted data
TB16.FTN	format data for Table #16	n/a	formatted data
TB17.FTN	format data for Table #17	data base information	formatted data
TB18.FTN	format data for Table #18	data base information	formatted data
TB20.FTN	format data for Table #20	data base information	formatted data
TB21.FTN	format data for Table #21	n/a	formatted data

Approximate Size: 16128 words

DATA BASE MODULE

Data Base Executive

Filename: DBEXEC.FTN

Driven by: UFEXEC.FTN

Description: This task handles all requests for data base information. A specially formatted Data Manipulation Language (DML) request is received and the appropriate information is retrieved for the task (or tasks) which must access the information.

Interface: Input - data base request (DML statement)
Output - disk-resident file of accessed information (DBFILE)

Associated Libraries: DBELB.OLB - Data Base Executive Library
F4PRMS.OLB - Fortran interface library for DEC's Record Management Services (RMS) Package
RMSLIB.OLB - RMS Library

Approximate Size: 512 words

Data Base Executive Library

Filename: DBELB.OLB

Driven by: DBEXEC.FTN

Description: This library supports the function of the Data Base Executive (DBEEC.FTN). Since it breaks down into three specific packages, they are presented individually following this discussion.

Supporting Packages:

TRANS.FTN - see Translate Package

IO.FTN - see Input/Output Package

OPS.FTN - See Operations Package

Translate Package - Data Base Executive Library Package

Filename: TRANS.FTN

Driven by: DBEXEC.FTN

Description: This package supports the translate function of the Data Base Executive (DBEXEC.FTN). It translates a data base request (DML statement) into a form necessary for the Input/Output Package.

Interface: Input - data base request (DML statement)
Output/Result - type of operation requested, file to be accessed, field names and values within a record that are to be operated on.

Supporting Routines:

Filename	Function	Input	Output/Result
GETCH.FTN	Returns the next character from the DML statement	DML statement	one character
TYPE.FTN	Determines the character type of Data Base request-query or update	the type condition	the type or
FLNME.FTN	Determines the file to be accessed - Airbase, Target, Mission	the filename character	the filename or error condition
DELIM.FTN	Determines if the character is a DML delimiter $, *, [,#,]	character	true or false

Filename	Function	Input	Output/Result
EOR.FTN	Determines if a character is an "end-of-request" character -]	character	true or false
EOS.FTN	Determines if a character is an "end-of-series" character of one or more DML statements-?	character	true or false
KAT.FTN	Builds the mneumonic field name	character	mnemonic field name
VAT.FTN	Builds the field value	character	field value

Approximate Size: 2560 words

Input/Output Package - Data Base Executive Library Package

Filename: IO.FTN

Driven by: DBEXEC.FTN

Description: This package supports the input/output function of the Data Base Executive(DBEXEC.FTN). It retrieves or updates the appropriate records from the data base which satisfy the mnemonic field names and values supplied by the translate package.

Interface: Input - (same as outputs for translate package)
Output/Result - data base records which satisfy request found in a disk-resident file (DBFILE.DAT)

Supporting Routines:

Filename	Function	Inputs	Outputs/Results
BAS.FTN	Determines if it is an update or query for airbase file	request type	invocation of BASQ or BASU
BASQ.FTN	Retrieves records from the airbase/unit files	field names and values	records retrieved and put in DBFILE.DAT
BASU.FTN	Updates records in the airbase/unit files	field names and values	records updated and rewritten to the airbase/unit files
TRGT.FTN	Determines if it is an update or query for a target file	request type	invocation of TRGTU or TRGTQ
TRGTQ.FTN	Retrieves records from a target file	field names and values	records retrieved and put in DBFILE.DAT

Filename	Function	Input	Output/Result
TRGTU.FTN	Updates target records	field names and values	records updated and rewritten to the target files
MISN.FTN	Determines if it is an update or query for the mission file	request type	invocation of MISNQ or MISNU
MISNQ.FTN	Retrieves records from the mission file	field names and values	records retrieved and put in DBFILE.DAT
MISNU.FTN	Updates/creates mission records	field names and values	records updated or created in the mission file
CID.FTN	updates the identify field of a target record as well as the incore array	target record	updated record and incore array
MPTR.FTN	updates the mission pointer field in a target record	target record	updated record
ACTW.FTN	updates the available aircraft during the given time window	unit record	updated record
CSNM.FTN	updates the call sign number at the given unit	unit record	updated record
ASN.FTN	updates the assignment indicator in the target record	target record	updated record
CREATE.FTN	builds a place holder for a newly created mission	mission number	place holder for record

Filename	Function	Input	Output/Result
IDENT.FTN	Identifies a target by altering the id field in the incore array	index into the incore array	target identified
UDENT.FTN	Unidentifies a target by altering the id field in the incore array	index into the incore array	target unidentified
RI.FTN	retrieves airbases from BASE1.DAT	base name	airbase record
OT.FTN	retrieves airbases with a given ordnance type	ordnance type	airbase record(s)
AT.FTN	retrieves airbases with a given aircraft type	aircraft type	airbase record(s)
GRPH.FTN	determines if it is a graphics update	request type	invocation of GRPHU
GRPHU.FTN	updates colors in incore array for graphic display	symbol names and colors	incore array updated
EOB.FTN	determines if a symbol is desired order of battle	symbol name, order of battle type	true or false

Approximate Size: 12544 words

Operations Package - Data Base Executive Library Package

Filename: OPS.FTN

Driven by: DBEXEC.FTN

Description: This library supports the data base operations of the Data Base Executive (DBEXEC.FTN). It is a set of general purpose data base functions (e.g., open file, reads, writes).

Interface: Input - data base operation request
Output/Result- data base operation

Supporting Routines:

Filename	Function	Inputs	Outputs/Results
INIT.FTN	performs initialization of queues and variables in the Translate Package	queues and variables	queues and variables set to their initial values
CLEAR.FTN	resets queues and variable in the Translate Package	queues and variables	queues and variables reset
GETALL.FTN	read all records of a given file	filename	records retrieved
WRITE.FTN	writes records to DBFILE.DAT	record	DBFILE.DAT populated
APND.FTN	adds the record address and graphics id to a target record prior to its being written to DBFILE	Record address and graphics id	Record appended
RDALL.FTN	reads all identified targets	n/a	identified target records read
QREM.FTN	removes a field name or value from the queue	queue	field name or value

Filename	Function	Input	Output/Result
FNAME.FTN	determines the filename (e.g., AAA.DAT, TNK.DAT) from the incore array	index into the incore array	filename
IDEC.FTN	converts a byte array containing an integer into its integer equivalent	array and array length	integer
OPENF.FTN	opens the given sequential file with direct access	filename	desired file file opened
GETKV.FTN	retrieves the field name and its associated value (via QREM's)	queues and lengths	fieldname and value
QADDB.FTN	adds an item to a queue	queue and length	new queue
TRNSFR.FTN	transfers one file into DBFILE	number of records to be transferred	records put in DBFILE
CPY.FTN	copies all airbase records to work area	n/a	work area populated
CLNUP.FTN	deletes old work files	n/a	garbage collection

Approximate Size: 3584 words

DECISION AID MODULE

Decision Aid Executive

Filename: DAEXEC.FTN

Driven by: UFEXEC.FTN

Description: This task is responsible for all decision aid requests. When a decision aid request is received and interpreted, the appropriate decision aid is executed.

Interface: Input - decision aid request
Output/Result - decision aid function

Associated Library: DAELB.OLB - Decision Aid Executive Library

Approximate Size: 98 words

Decision Aid Executive Library

Filename: DAELB.OLB

Driven by: DAEXEC.FTN

Description: This library supports the function of the Decision Aid Executive (DAEXEC.FTN). The collection of decision aids available with CEDA is accessed through this library.

Supporting Routines:

Filename	Function	Input	Output/Result
DINTRP.FTN	interpret decision aid request	decision aid request	decision aid function code
WEAPCH.FTN	select weapon for allocation	target choice	selected weapons list
THRTST.FTN	determine threats within the target area	target location	threats in area
DIST.FTN	calculate distance between points	location of points	distance
WTWOP.FTN	assign weights to weapon choices from the AWOP file	weather and target information	weighted choices
BADWTH.FTN	weight for bad weather choice	aircraft	weighted information
FRWTH.FTN	weight for fair weather choice	aircraft	weighted information
EMIT.FTN	weight for emitter target choice	aircraft	weighted and ordnance information

Filename	Function	Input	Output/Result
SAMTH.FTN	weight for SAM threats choice	aircraft	weighted information
AIRTH.FTN	weight for AIR threats choice	aircraft	weighted information
AAATH.FTN	weight for AAA threats choice	aircraft	weighted information
SORT.FTN	sort choices in order of weights	weights	sorted choices
SORTIE.FTN	calculate number of sorties required for each choice	probability of damage and aircraft information	sorties
GETORD.FTN	from the SCL file, get the number and type of ordnance required for each choice	standard configuration load code	number and type of ordnance
DBWRIT.FTN	output weapon choices to DBFILE for further access	best weapon choices	data file containing best weapon choices
BASECH.FTN	select airbases for allocation	target and weapon choice and full airbase list	selected airbase list
GETBS.FTN	from DBFILE, retrieve the base record for each airbase	record identification	base record

Filename	Function	Input	Output/Result
ICKOR.FTN	check if ordnance is available at base and in sufficient quantity	number and type of ordnance	availability
ICKAC.FTN	check number of aircraft in a unit at base	aircraft information for weapon	aircraft available and optimum choice TOT for unit
ABPRI.FTN	prioritize airbases according to certain criteria	base information	sorted bases
RANK.FTN	rank elements of an array according to maximum or minimum values	array and code to select	sorted array maximum or minimum ordering
DBBASE.FTN	output base choices to DBFILE for further access	base choices	data file containing base choices
MISSN.FTN	generate a mission choice	target, weapon, unit and base choice from DBFILE	proposed mission
GETTAR.FTN	from DBFILE retrieve target record	record identification	target record
MSNTYP.FTN	determine mission type	target type	mission type
GETAB.FTN	retrieve airbase record from DBFILE	record identification	airbase record
GETWEP.FTN	from DBFILE retrieve weapon record	record identification	weapon record

Filename	Function	Input	Output/Result
GETUNT.FTN	from DBFILE retrieve unit record	record identification	unit record
CALLSN.FTN	determine the call sign	unit call name and number	call sign
DBMSN.FTN	output mission to DBFILE for further access	mission choice	data file containing mission choice
ATO.FTN	set up for ATO function	n/a	requests to satisfy ATO function
HCTB20.FTN	output hardcopy ATO	ATO record	hardcopy ATO from DBFILE

Approximate Size: 44032 words

General Library

Filename: GENLB.OLB

Driven by: All CEDA tasks and other libraries may access this library.

Description: This library supports functions which are common to several (or all) of the CEDA modules.

Supporting Routines:

Filename	Function	Input	Output/Result
HSET.FTN	lock/unlock keyboards	terminal number and lock/unlock request	locked/unlocked keyboard
SWITCH.FTN	switch brightness brightness on/off	terminal number and on/off request	on/off
PULSE.FTN	pulse a channel with on/off list	channel number and on/off request	on/off pulse
BOOKPR.FTN	search array for a particular value	array and value	location in array
QADD.FTN	add entries to an array (beginning or end)	array and new entries	new array
QREM.FTN	remove entries from an array (beginning or end)	array and number of entries to remove	new array
AIRCOD.FTN	converts all aircraft type code into all ASCII string	aircraft type code	ASCII string

Filename	Function	Input	Output/Result
ORDCOD.FTN	converts all ordnance type code into all ASCII string	ordnance type code	ASCII string
TARCOD.FTN	converts a target type code into an ASCII string	target type code	ASCII string
RSET.FTN	reset selection criteria and airbase prioritization at end of each allocation process	n/a	reset allocation factors

Approximate Size: 4864 words

SECTION 4

FILES DESCRIPTION

4.1 GENERAL

This section describes the disk-resident files which support CEDA. The discussion is divided into the following classes of files: Data Base Files which support the data base, Decision Aid Files which contain weaponeering files and the Geographic Files which support the graphics of CEDA. Appendix E contains the data dictionary and actual record layouts for all of the files.

4.2 DATA BASE FILES

4.2.1 File Creation

The target files, friendly airbase file and unit file were created from the files used by the Tactical Operations Planner (TOP) Phase 2 (1). The original files were converted from the format necessary for UNIX (on which TOP was run) to RSX format. The RMS files were created by using an RMS utility ($RMSDEF) and populated by a Fortran program. The Fortran program chose fields from the original files selectively so that the records in the CEDA files are a subset of the TOP records.

4.2.2 Target Files

There are 113 target records stored in 13 target files according to generic target type as follows:

AAA.DAT	-	anti-aircraft artillery targets
ART.DAT	-	artillery battery targets
EW.DAT	-	early warning type targets
ROC.DAT	-	rockets
FLD.DAT	-	enemy airbases
RAD.DAT	-	radar targets
BRI.DAT	-	bridge targets
TRK.DAT	-	truck targets
TNK.DAT	-	tank targets
MIL.DAT	-	military type targets (barracks, depot, etc.)
POL.DAT	-	petroleum, oil and lubricant type targets

IND.DAT - industrial type targets
SAM.DAT - surface to air missiles

Each record is 256 bytes in length. The first 178 bytes of each record contains the same type of information regardless of target type. The type of information is pertinent to targets in general such as target name, BE number, weather ceiling, etc., while the remainder of each record is target-type specific such as construction for bridges, surface condition for runways and current amount for petroleum, oil and lubricant type targets. (See Appendix E for record layouts of each file).

Each target file is organized sequentially and is accessed directly (via the record's position in the file) in most cases. Target records are retrieved through an incore array which lists the filename and record address for each target symbol on the graphics screen.

4.2.3 Friendly Airbase File - BASE1.DAT

There are a total of twelve friendly airbases contained in BASE1.DAT. Each record is 256 bytes in length and, in addition to other information, includes the number of units (or squadrons) located at the base and the record address of the first unit record found in UNIT.DAT.

This file is organized as an indexed file and accessed by keying on the primary key, ICAO name, which is located in bytes 1-4.

4.2.4 Unit File - UNIT.DAT

A unit represents a squadron resident at a particular airbase. There are 67 unit records, each 256 bytes in length. UNIT.DAT is organized sequentially with all units belonging to a particular base stored contiguously. The airbase file, BASE1.DAT, contains the address of its first unit and the total count. This file, therefore, is accessed directly via the record's position in the file.

4.2.5 Mission File - MISION.DAT

MISION.DAT is created while running CEDA. It contains the following types of missions:

Proposed - Missions that have been created by the allocation process and accepted during the last phase.

Confirmed - Missions that have been confirmed via the Planned Missions Menu.

Fragged - Missions that have been sent to wing via the ATO function.

Note that the union of proposed missions and confirmed missions constitute planned missions.

The mission file is organized as an indexed file and records of 256 bytes in length are accessed by keying on one of the following keys:

Type	Name	Location	Purpose
Primary Key	MN	1-3	Mission Number
Alternative Key 1	ME	58	Mission State
Alternative Key 2	MF	90	Fragged Flag

4.2.6 Work Area File - DBFILE.DAT

This file is a disk-resident "work area". The results of all data base queries are recorded in this file as well as the results of the decision aids. DBFILE.DAT contains records 256 bytes in length and is organized sequentially. Access to this file can be either sequential or direct access by specifying the record's address.

The first record of DBFILE contains header information indicating the number and type of records (specified by their order) as follows:

NT - number of target type records or symbol (Graphic Options) type records
NM - number of mission type records
NA - number of friendly airbase type records
NU - number of unit type records
NW - number of weapon choice type records
NB - number of base choice type records
NSM - number of suggested mission type records
NN - total number of records found in DBFILE

The record layouts for all the above record types can be found in Appendix E.

4.2.7 Data Dictionary File - CODE.DAT

This file contains the mnemonic field name, field starting position and field length for all fields in the target files, BASE1.DAT, UNIT.DAT and MISION.DAT.

Each record is 8 bytes in length with keyed access on the field name bytes 1-4. Organization is indexed.

This arrangement allows the record layouts to change thereby only affecting CODE.DAT and not the data base routines.

4.3 DECISION AIDS FILES

4.3.1 File Creation

The weaponeering file, AWOP.DAT, and the standard configuration load codes file, SCL.DAT, contain the information found in the corresponding TOP (1) files. They were populated by using DATATRIEVE to create the empty files and then typing in the record information from hard copies of the TOP files.

4.3.2 Weaponeering File - AWOP.DAT

This file contains 637 records with 15 bytes of information padded on the right to form records 16 bytes in length. Each record contains weaponeering information such as aircraft type, SCL code, dive angle, intevelometer, altitude, airspeed and probability of damage pertinent to a particular generic target type.

AWOP.DAT is organized sequentially and can support both sequential and direct access. To access directly, a table is stored in a common area that contains the record address of the first AWOP record for a particular generic target type. AWOP records for a specific generic target type are stored in sequential order.

Appendix E gives the record layout for AWOP.DAT.

4.3.3 Standard Configuration Load Code File - SCL.DAT

This file contains information about the type and number of ordnance contained in a specific load. SCL.DAT is organized and accessed sequentially. It contains records 32 bytes in length with the first byte containing the load code. The remaining record contains the number and type of primary ordnance and up to three secondary ordnances. See Appendix E for the record format of SCL.DAT.

4.4 GEOGRAPHIC DATA FILES

4.4.1 File Creation

The national boundaries in the geographic data files for CEDA were generated on a digitizer tablet using a stereographic projection of Western Europe. All symbols and cities were translated from their known latitude/longitude positions to their corresponding (pixel) position within the geographic area defined above.

4.4.2 Geographic Area Files - AR01.DAT - AR21.DAT

The area files contain information necessary to graphically display political boundaries and foreground information. These files vary in number of records, however each record contains 80 bytes and they are organized sequentially. See Appendix E for the record layout of these files.

SECTION 5

SYSTEM FLOW DESCRIPTION BY MODULE

This section presents the high level flow of processing within each of the functional modules of CEDA defined above. Since it is only intended to give an overall view, the flow descriptions included here reference routines which were defined in Section 3, "Software Description". Figure 5-1, shows the breakout of memory-resident and disk-resident tasks, libraries and Common regions. It also shows the relationship between each of the functional modules. To complete the CEDA system flow description, Figures 5-2 through 5-9 show flow diagrams for each executive task.

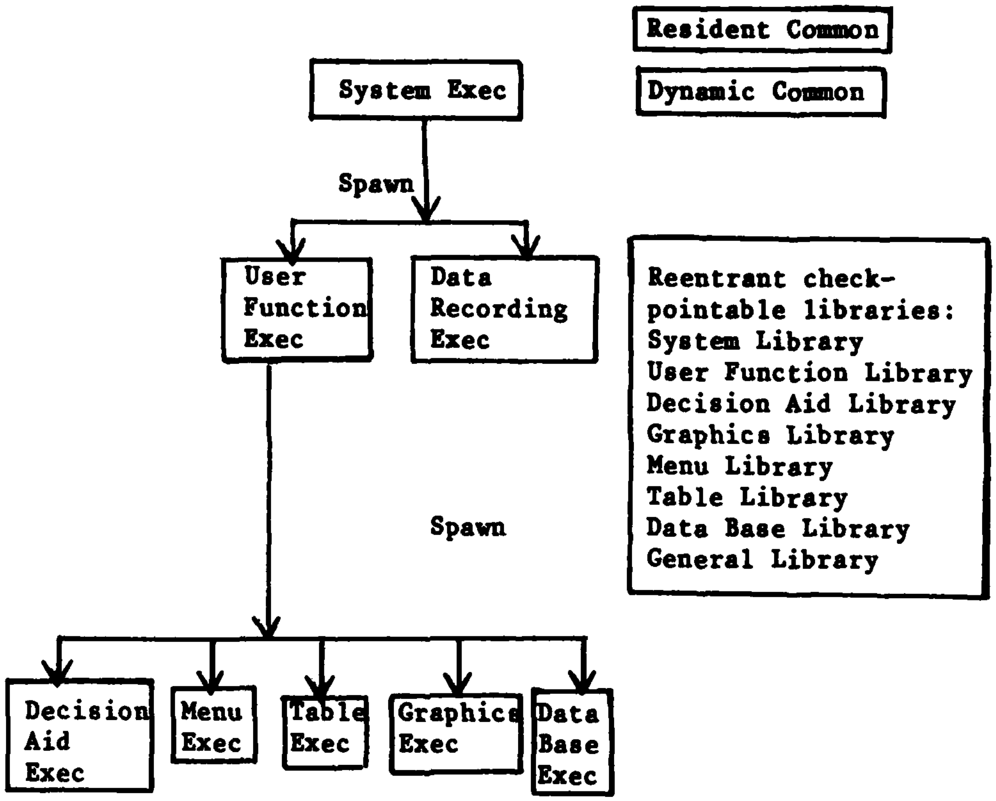


Figure 5-1 System Flow

System Executive - SEXEC.FTN

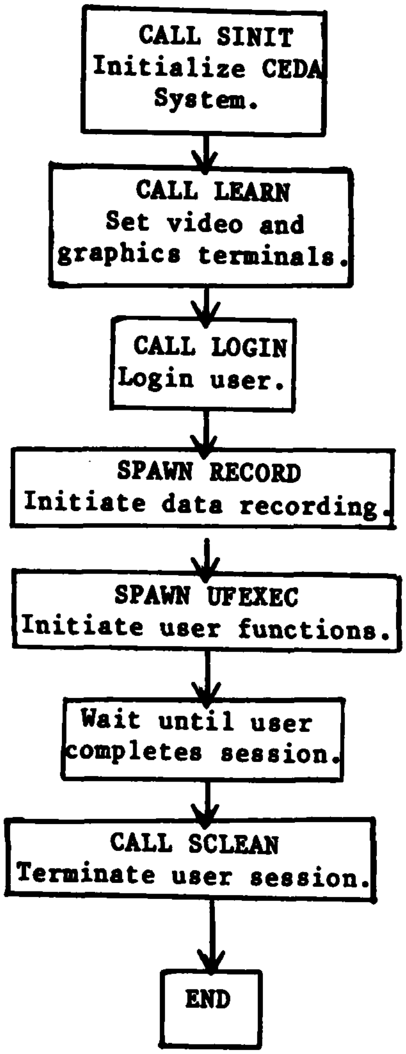


Figure 5-2 System Executive

Data Recording Executive - RECORD.FTN

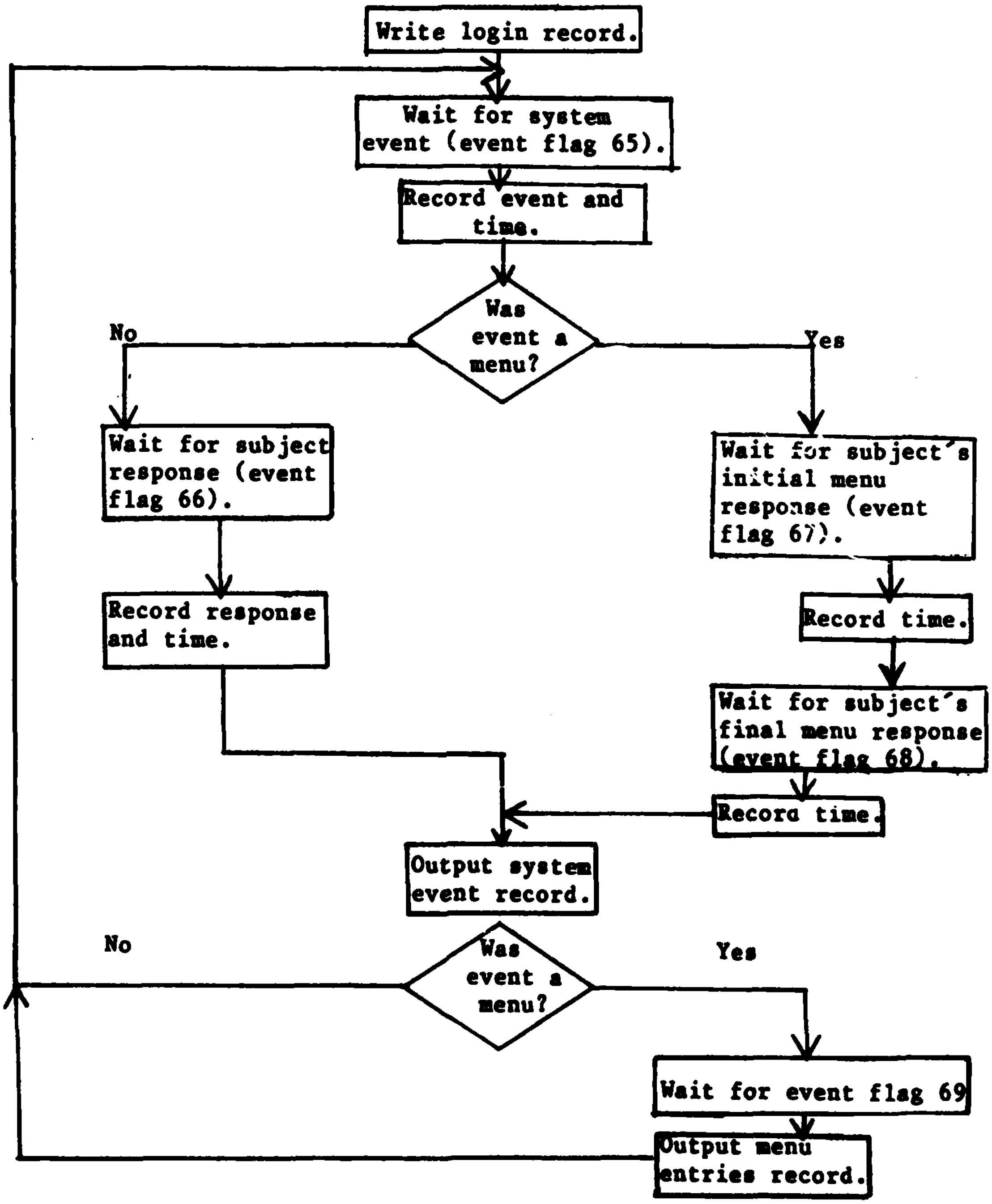

Write login record.
Wait for system event (event flag 65).
Record event and time.
Was event a menu?
No
Yes
Wait for subject response (event flag 66).
Record response and time.
Wait for subject's initial menu response (event flag 67).
Record time.
Wait for subject's final menu response (event flag 68).
Record time.
Output system event record.
Was event a menu?
No
Yes
Wait for event flag 69
Output menu entries record.

Figure 5-3 Data Recording Executive

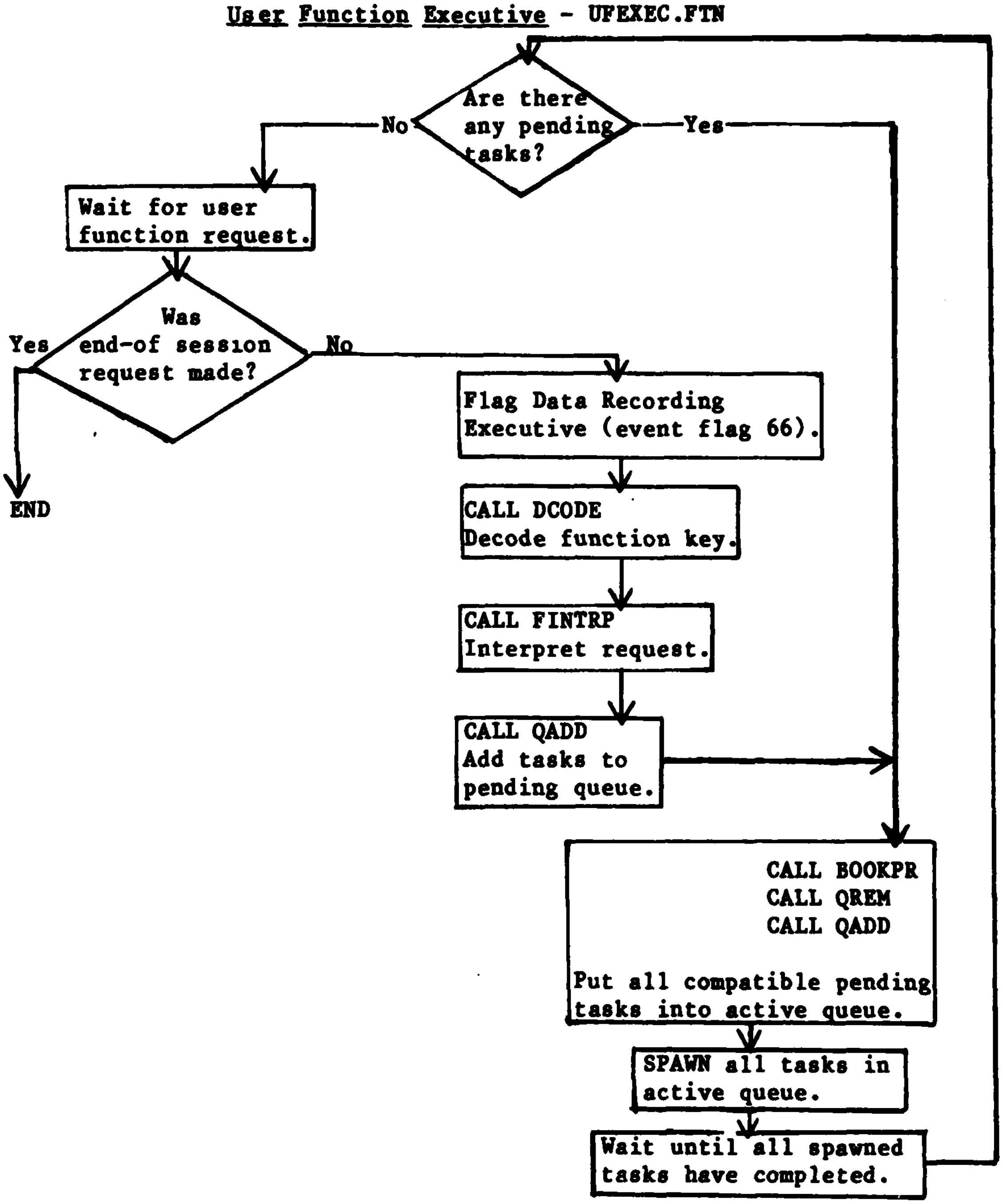

User Function Executive - UFEXEC.FTN
Are there any pending tasks?
No
Yes
Wait for user function request.
Was end-of session request made?
Yes
No
END
Flag Data Recording Executive (event flag 66).
CALL DCODE Decode function key.
CALL FINTRP Interpret request.
CALL QADD Add tasks to pending queue.
CALL BOOKPR
CALL QREM
CALL QADD
Put all compatible pending tasks into active queue.
SPAWN all tasks in active queue.
Wait until all spawned tasks have completed.

Figure 5-4 User Function Executive

Graphics Executive - GEXEC.FTN

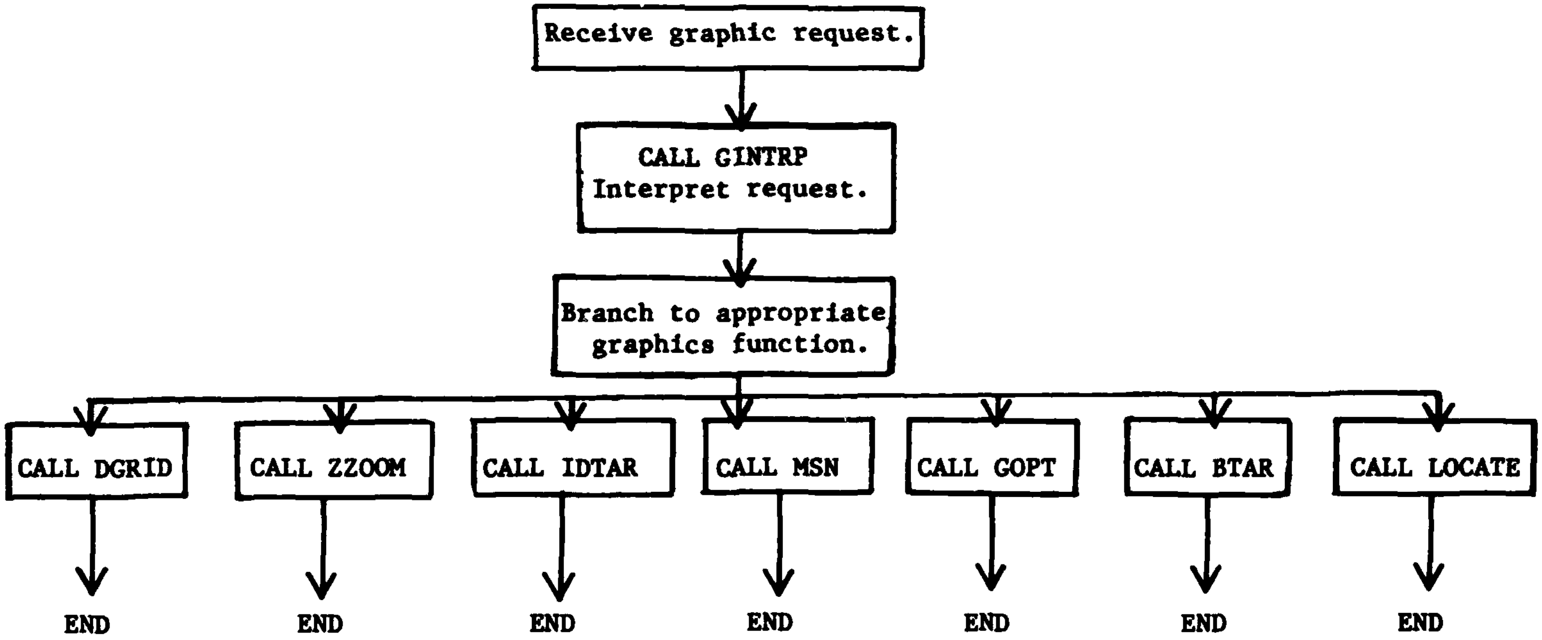


Figure 5-5. Graphics Executive

Menu Executive - MEXEC.FTN

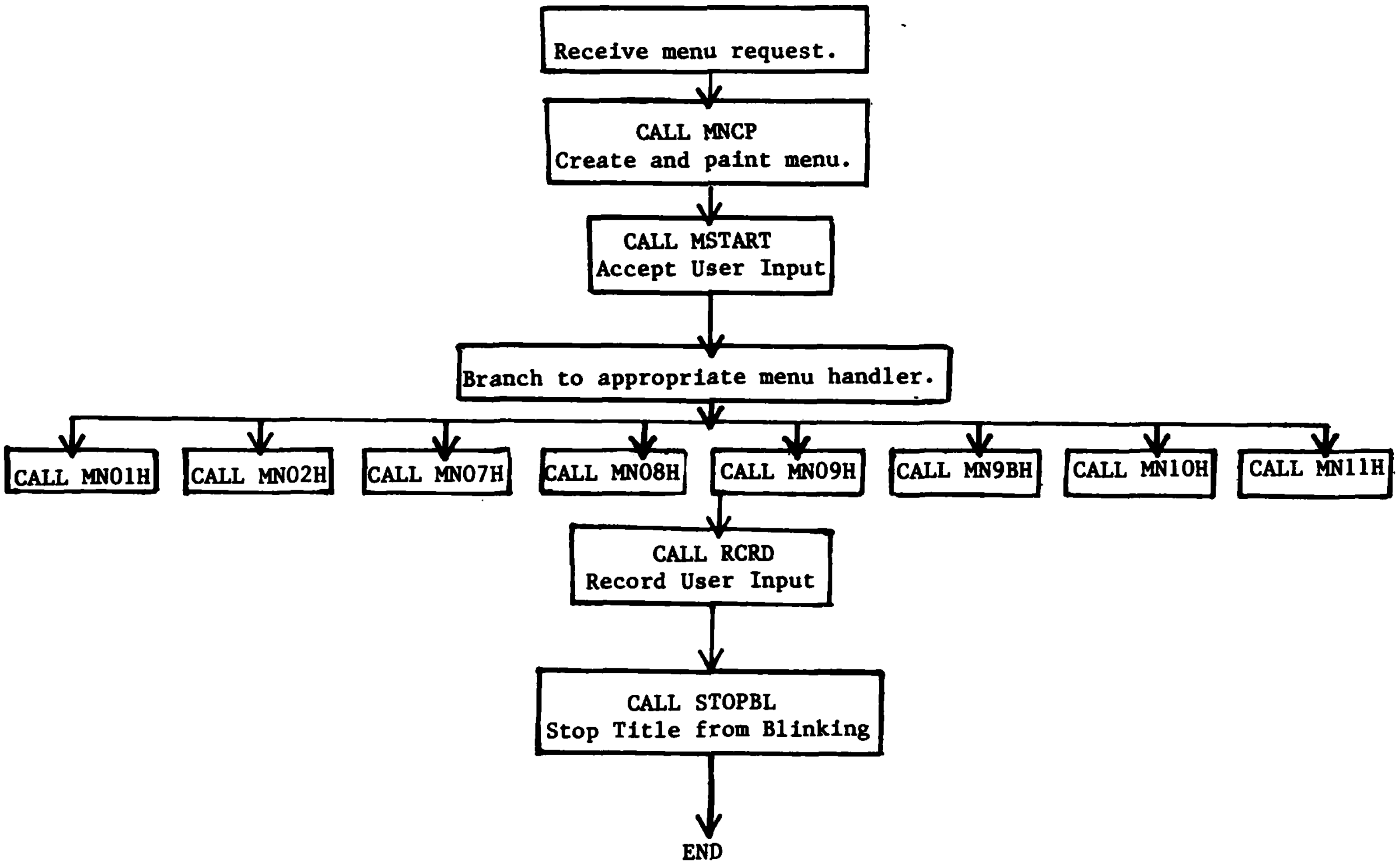


Figure 5-6. Menu Executive

Table Executive - TEXEC.FTN

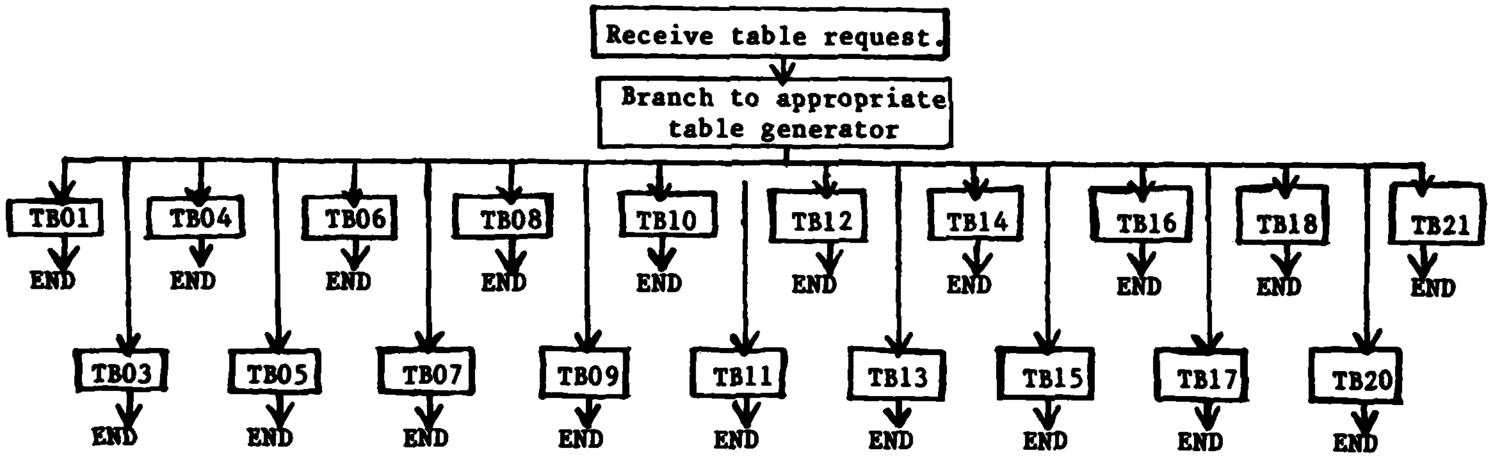


Figure 5-7. Table Executive

__Data Base Executive__ - DBEXEC.FTN

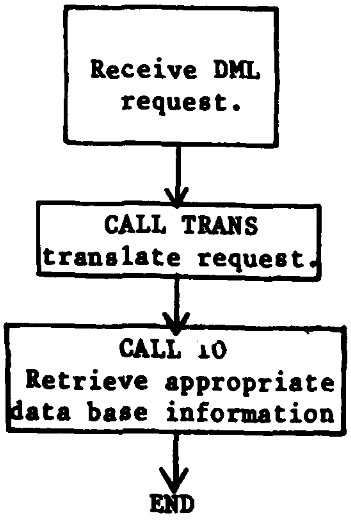

Receive DML request.
CALL TRANS translate request.
CALL IO Retrieve appropriate data base information
END

Figure 5-8 Data Base Executive

Decision Aid Executive - DAEXEC.FTN

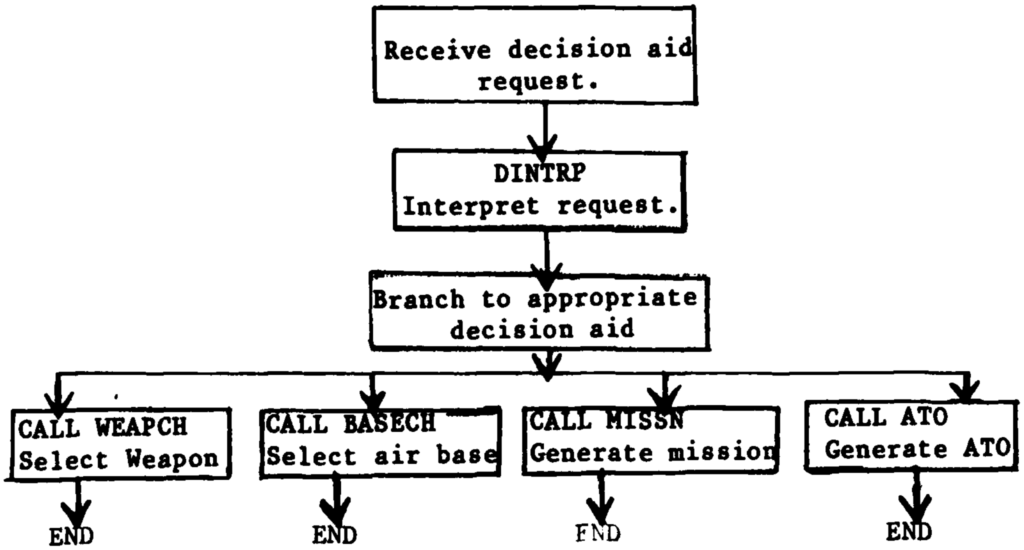


Figure 5-9. Decision Aid Executive

SECTION 6

HARDWARE SYSTEM USED FOR DEVELOPMENT

As previously stated, CEDA was developed on the Command and Control Concept Evaluation Capability (C^2 CONCAP) facility at MITRE and took advantage of other efforts in the area of automated C^2 systems. This facility was developed under Project 4960 and has as its primary goal to provide planners with general capabilities for selecting, analyzing, evaluating, and refining system concepts. CEDA is a tool for experimenting with C^2 decision making via an online, interactive system.

The Human Resources Laboratory (HRL) at Wright Patterson Air Force Base has the following configuration which makes it possible to transfer and run CEDA at that facility also.

Processor and Memory

- PDP-11/44 central processor with 8KB cache memory and memory management
- FP11-F Floating point processor
- 512 kilobytes ECC MOS memory
- Bootstrap module with diagnostic, line frequency cl´ck
- KW11-P programmable real-time clock

Peripherals

- RJMO2 disk controller and RMO2 67 megabyte disk drives
- RL211 disk controller and two 10.4 megabyte RLO2 removable disk cartridge drives
- RX211-BA floppy disk controller and two RXO2 .5 megabyte drives
- TU58-DA cabinet mounted dual transport cartridge tape subsystem
- LA120-DA Decwriter 3 printing terminal (console terminal)

- TS11-CA single drive, nine track, 1600 bit per inch magnetic tape subsystem

- LP11-AA 300 line per minute, 64-character ASCII band printer

- Trilog Colorplot 100 printer/plotter

Communications Interfaces and Terminals

- Two DL11 EIA/CCITT asynchronous serial line interfaces (for the LA120 console terminal and the TU58 cassette tape)

- DR11-B direct memory access interface

- DH11-AD programmable, asynchronous 16-line multiplexer

- DZ11-A asynchronous, full-duplex, eight-line multiplexer with programmable speeds up to 9600 bits per second

- DRS11-A 48 bit output module

- Tektronix 4027 color-graphics terminal

- Two DEC VT100 alpha-numeric terminals

Ancillary System

- Aydin High Resolution Color-Graphics System, includes: model 5216 display computer

- Keyboard and trackball

- Digital Equipment RK05 2.4 megabyte cartridge disk drive

APPENDIX A

SYSTEM EVENT CODES

Possible values for system events are:

Menus:	'MN01'	Graphic Options Menu	
	'MN02'	Data Base Search Menu	
	'MN07'	Weapons Allocation Menu	
	'MN08'	Interactive Allocation/Weapon Decision Menu	
	'MN09'	Interactive Allocation/Airbase Decision Menu	
	'MN9B'	Interactive Allocation/Mission Decision Menu	
	'MN10'	Planned Missions Menu	
	'MN11'	Airbase Prioritization Menu	
Tables:	'TB01'	Target Description	
	'TB03'	Identified Targets Summary	
	'TB04'	Friendly Airbase Description	
	'TB05'	Friendly Airbases	
	'TB06'	Mission Schedule	
	'TB07'	Ground Order of Battle	
	'TB08'	Air Order of Battle	
	'TB09'	Enemy Order of Battle/Electronic	
	'TB10'	Enemy Order of Battle/Threat	
	'TB11'	Fighter Schedule	
	'TB12'	Alert Aircraft	
	'TB13'	Command and Control Elements	
	'TB14'	Target Type	
	'TB15'	Ordnance Load Codes	
	'TB16'	Aircraft Characteristics	
	'TB17'	Identify Line/Targets	
	'TB18'	Identify Line/Friendly Airbase	
	'TB20'	Air Tasking Order	
	'TB21'	Continue Session	
Graphics:	'AR01'	Geographic Area #1	total area
	'AR02'	Geographic Area #2	first quadrant
	'AR03'	Geographic Area #3	second quadrant
	'AR04'	Geographic Area #4	third quadrant
	'AR05'	Geographic Area #5	fourth quadrant
	'AR06'	Geographic Area #6	first quadrant of Area #2
	'AR07'	Geographic Area #7	second quadrant of Area #2
	'AR08'	Geographic Area #8	third quadrant of Area 2
	'AR09'	Geographic Area #9	fourth quadrant of Area #2
	'AR10'	Geographic Area #10	first quadrant of Area #3
	'AR11'	Geographic Area #11	second quadrant of Area #3
	'AR12'	Geographic Area #12	third quadrant of Area #3

'AR13' - Geographic Area #13 - fourth quadrant of Area #3
'AR14' - Geographic Area #14 - first quadrant of Area #4
'AR15' - Geographic Area #15 - second quadrant of Area #4
'AR16' - Geographic Area #16 - third quadrant of Area #4
'AR17' - Geographic Area #17 - fourth quadrant of Area #4
'AR18' - Geographic Area #18 - first quadrant of Area #5
'AR19' - Geographic Area #19 - second quadrant of Area #5
'AR20' - Geographic Area #20 - third quadrant of Area #5
'AR21' - Geographic Area #21 - fourth quadrant of Area #5

APPENDIX B

USER RESPONSE CODES

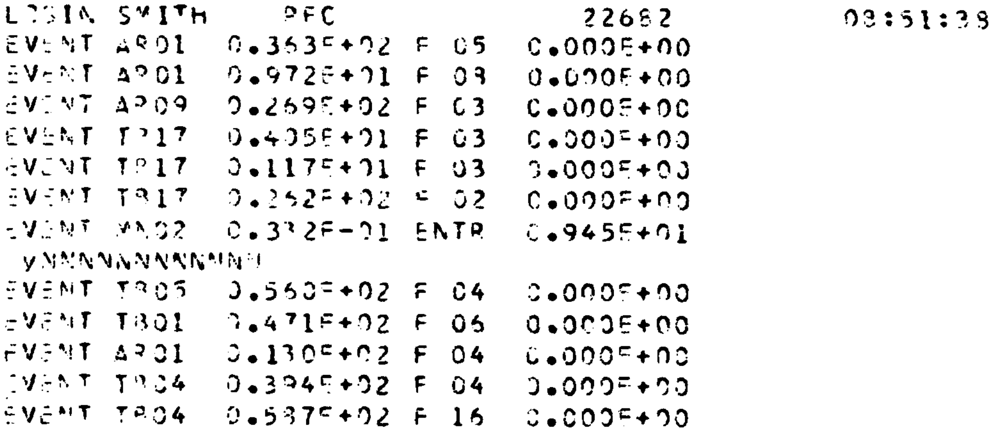
Possible values for user response are:

'F01'	-	Function Key #1
'F02'	-	Function Key #2
'F03'	-	Function Key #3
'F04'	-	Function Key #4
'F05'	-	Function Key #5
'F06'	-	Function Key #6
'F07'	-	Function Key #7
'F08'	-	Function Key #8
'F09'	-	Function Key #9
'F10'	-	Function Key #10
'F11'	-	Function Key #11
'F12'	-	Function Key #12
'F13'	-	Function key #13
'ENTR'	-	Menu Entry

The function keys are set prior to session initiation. Refer to the CEDA User's Manual for information on how they are set for the initial system.

APPENDIX C

SAMPLE DATA COLLECTION



```
LOGIN SMITH    PFC                 22682           08:51:38
EVENT AR01  0.363E+02 F 05  0.000E+00
EVENT AR01  0.972E+01 F 08  0.000E+00
EVENT AR09  0.269E+02 F 03  0.000E+00
EVENT TR17  0.405E+01 F 03  0.000E+00
EVENT TR17  0.117E+01 F 03  0.000E+00
EVENT TR17  0.262E+02 F 02  0.000E+00
EVENT MN02  0.332E-01 ENTR  0.945E+01
 YNNNNNNNNNNNN
EVENT TR05  0.560E+02 F 04  0.000E+00
EVENT TR01  0.471E+02 F 06  0.000E+00
EVENT AR01  0.130E+02 F 04  0.000E+00
EVENT TR04  0.394E+02 F 04  0.000E+00
EVENT TR04  0.537E+02 F 16  0.000E+00
```

APPENDIX D

DATA MANIPULATION LANGUAGE (DML)

The Data Manipulation Language is a data base query language developed at MITRE originally to support the data base of the TOP (Tactical Operations Planner) system. It was altered to better suit the needs of CEDA.

This appendix includes the rules for forming a DML statement and the statements that are currently available with CEDA including when they are used.

RULES

General Format DML Statement

TF[R]

T - type of request

Q - query
U - update

F - file to be accessed

A - friendly airbase file
T - target file
M - mission file

R - the request

General Request Form

MNV*#UP#UV*

MN - mnemonic code for field name within a record
V - value of field corresponding to MN
UP - mnemonic code for field to be updated
UV - updated value

Rules

1. Character #1 must be of type T.
2. Character #2 must be of type F.
3. A single request is enclosed in a pair of square brackets.
4. A mnemonic code is surrounded by two dollar signs ($).
5. A field value or update value is terminated with an asterisk (*).
6. An update mnemonic is surrounded by pound signs (#).
7. There may be only one update mnemonic and associated update value per request. (#UP#UV*)
8. There may be as many mnemonic codes and values as needed. (MNV*)
9. Update mnemonics and values are invalid with requests of type Q.
10. Empty square brackets [ ] is a valid request with a request of type Q and indicates retrieval of all records in the file.
11. A series can contain more than one DML statement.
12. A series is terminated by a question mark (?).

AVAILABLE DML STATEMENTS

Target Queries

1.
```
QT [$GD$___*]
```

Meaning: Retrieve the target with the specified graphics id (GD).

Use:

1. When the full page descriptor key is hit, this query is generated with the graphics id of the target where the cursor was positioned.

2. When a target is identified via function key #3, this query is generated to retrieve the record for display in the Identify line (TB17).

2.
```
QT [$ID$ALL*]
QT [$ID$__*]
QT [$ID$__*$ID$__* ... $ID$___*]
```

Meaning: Retrieve the target(s) with the specified id number. If "ALL" is specified retrieve all identified targets.

Use: When the first phase of allocation is invoked, the user inputs "ALL" to allocate against all the identified targets or the specific target numbers. One of the above queries is then formed.

Target Updates

1.
```
UT [$GD$___*#ID#__*]
```

Meaning: Identify or unidentify the target with the specified graphics id.

Use: When the identify key is hit, this query is generated with the graphics id of the target where the cursor was positioned. If the target was previously identified the id number will be "00" generating an unidentify, otherwise the id number will be the next available one.

2. UT [GD___*#TM#___*]

Meaning: Alter the mission pointer of the target with the specified graphics id.

Use: 1. In the last phase of allocation where a mission is accepted, the target's mission pointer is set to the record address of the mission.

2. If the mission is rejected in the Planned Missions menu, the mission pointer in the target record is set to all zeros.

3. UT [GD___*#TA#_*]

Meaning: Alter the assignment indicator of the target with the specified graphics id.

Use: 1. In the last phase of allocation where a mission is accepted, the target's assignment indicator is set to the symbol "@".

2. If the mission is rejected in the Planned Missions menu, the assignment indicator in the associated target record is set to blank.

Airbase/Unit Queries

1. QA [GD___*]

Meaning: Retrieve the airbase and its associated unit records with the specified graphics id.

Use: 1. When the full-page descriptor key is hit, this query is generated with the graphics id of the airbase where the cursor was positioned.

2. When the identify key is hit with the cursor positioned on a friendly airbase, this query is generated.

2. QA []

Meaning: Retrieve all the friendly airbases and their associated unit records.

Use: When the first phase of allocation is invoked, if no selection criteria has been input to selectively retrieve airbases, all are chosen.

3. QA [RI____*RI____*...
OT_*
AT_*AT_* ...]

Meaning: Retrieve the correct airbases/units with the given ICAO (RI) name and the ordnance and aircraft types (OT, AT) available.

Use: From the first phase of allocation when selection criteria has been chosen, this query is generated. Not all of it is required, only fields input are used. For example, if only ordnance type was chosen - QA [OT_*]. Up to 5 possible RI's can be given and four possible aircraft types (AT).

<u>Airbase/Unit Updates</u>

1. UA [OU__*#UH#_______*]

Meaning: This query updates the number of aircraft available at the specified unit (OU) during the time window given. The seven digits following UH have the following meaning:

1-2 take-off time where 0800=08, 1300=13,etc.
3-4 return time
5 + increment, - decrement # of aircraft
6-7 # of aircraft

Use: 1. When a mission is accepted in the last phase of the allocator, this query is generated to decrement the number of aircraft available at the unit from take-off to return time.

2. When a mission is rejected in the Planned Missions Menu, this query increments the number of aircraft available at the unit from take-off to return time.

2. UA [OUS__*#CN#__*]

Meaning: Increments the call sign number so that all call sign numbers with the same tens digit are from the same mission. For example:

Mission #1	AC1	LASAR38
	AC2	LASAR39
	AC3	LASAR40
	AC4	LASAR41
Mission #2	AC1	LASAR50
	AC2	LASAR51
Mission #3	AC1	LASAR60

Use: When a mission is accepted in the last phase of allocation, this query is generated to increment the call sign number at the unit to reflect the number of aircraft flying the mission.

Mission Queries

1. QM [MF_*]

Meaning: Retrieve all the missions with a mission fragged flag equal to the value given.

Use: 1. From the last allocation phase, retrieves all planned missions (confirmed and proposed but not yet fragged) for the Planned Missions menu when the value equals zero.

2. From the Data Base Search menu when all scheduled missions are requested, this query is generated to retrieve all fragged missions with the value equals one.

2. QM [MN___* ... MN...*]

Meaning: Retrieve the mission(s) with the given mission number(s).

Use: Required to retrieve a mission or missions for display or removal of allocation pairing lines.

3. QM [ME_*]

Meaning: Retrieve all the missions with the given mission state.

Use: 1. From the ATO key to retrieve all the confirmed missions to send to wing when the value equals C.

2. From the Graphic Options Menu to display or remove allocation pairing lines.

Mission Updates

1. UM [$MN$OOO*#MN#___*]

Meaning: Create a new mission record.

Use: When a mission is accepted in the last phase of allocation, this query is generated to build a place holder for the newly created mission.

2. UM [$MN$___*#XX#...*]

Meaning: Alters the field specified for the mission records with the given mission number where xx is one of the field mneumonics for a mission record. (See Appendix E.)

Use: 1. When a mission is accepted after allocation, the new mission must be added to the mission file.

2. On the Planned Missions menu when the mission state (XX=ME) is either confirmed or rejected, this query is generated.

3. On the Planned Missions menu when a mission is to be displayed or removed (XX = M1, display flag) this query is generated.

3. UM [$ME$_*#MF#_*]

Meaning: Alter the fragged mission flag for missions of the given state.

Use: When the ATO key is hit, all confirmed.

4. UM [$ME$_*#M1#_*]

Meaning: Alter the value of the display flag for all confirmed missions.

Use: From the Graphic Options Menu, to indicate if allocation pairing lines are to be displayed or removed.

Graphic Update

1. UG [$NT$ALL*#CR#__*]

Meaning: Alter the color of all nominated targets.

Use: From the Graphic Options Menu when a user requests the color of nominated targets to change (00-06) or be removed (07).

2. UG [TE_*#CR#--*]

Meaning: Alter the color of all targets of a particular order of battle type where TE = G (ground), TE = A (Air), TE = E (Electronic), TE = S (Sam/AAA).

Use: From the Graphic Options Menu when a user requests the color of specific order of battle type targets to change (00-06) or not be displayed (07).

3. UG [$ID$ALL*#CR#__*]

Meaning: Alter the color of all identified targets.

Use: From the Graphic Options Menu when a user requests the color of identified targets to change (00-06) or be removed (07).

4. UG [FAALL*#CR#--*]

Meaning: Alter the color of all friendly airbases.

Use: From the Graphic Options Menu where the user requests the color of friendly airbases to change (00-06) or be removed (07).

5. UG [$CC$ALL*#CR#__*]

Meaning: Alter the color of all command and control centers.

Use: Where the user requests the color of command and control centers to change (00-06) or be removed (07), from the Graphic Options Menu.

6. UG [SRALL*#CR#__*]

Meaning: Display or remove all sam rings.

Use: From the Grpahic Options Menu, when the user requests sam rings to be displayed (00) or removed (07).

APPENDIX E

DATA DICTIONARY AND RECORD LAYOUTS

DATA DICTIONARY

Field Name	File Name	Starting Byte	Length
AA1-AA10	all target files	72,83,94 105,116,127 138,149,160,171	3
AC1-AC5	BASE1.DAT	58,63,68,73,78	4
AD1-AD10	all target files	68,79,90,101, 112,123,134,145, 156,167	1
AF1-AF10	all target files	69,80,91,102 113,124,135,146 157,168	1
AG	AAA.DAT	179	1
AI1-10	all target files	71,82,93,104,115 126,137,148,159,170	1
AK1-10	all target files	70,81,92,103,114 125,136,147,158,169	1
AN1-10	all target files	66,77,88,99,110 121,132,143,154,160	1
AO1-10	all target files	67,78,89,100,111, 122,133,144,155,166	1
AP1-10	all target files	75,86,97,108,119, 130,141,152,163	2
AT1-10	BASE1.DAT	57,62,67,72,77	1
AV	ART.DAT	179	1
AP1-10	all target files	75,86,97,108,119, 130,141,152,163	2
BC	BRI.DAT	180	1
BL	BRI.DAT	181	5
BT	BRI.DAT	179	1
BW	BRI.DAT	186	5
CN	UNIT.DAT	238	2
ER	EW.DAT	180	1
ET	EW.DAT	179	1
IC	IND.DAT	180	1
IL	IND.DAT	181	3
IT	IND.DAT	179	1
IW	IND.DAT	184	3
MA	MISION.DAT	15	3
MB	MISION.DAT	30	10

Field Name	File Name	Starting Byte	Length
MC	MISION.DAT	18	1
MD	MISION.DAT	43	2
ME	MISION.DAT	58	1
MF	MISION.DAT	90	1
MG	MISION.DAT	50	4
MH	MISION.DAT	11	4
MI	MISION.DAT	20	4
MJ	MIL.DAT	179	1
MK	MISION.DAT	7	4
ML	MISION.DAT	83	4
MM	MISION.DAT	59	20
MN	MISION.DAT	1	3
MO	MISION.DAT	19	1
MP	MIL.DAT	180	5
MQ	MISION.DAT	45	2
MR	MISION.DAT	47	3
MS	MISION.DAT	24	6
MT	MISION.DAT	40	3
MU	MISION.DAT	54	4
MV	MISION.DAT	79	2
MW	MIL.DAT	185	5
MX	MISION.DAT	81	2
MY	MISION.DAT	4	3
MZ	MISION.DAT	87	3
M1	MISION.DAT	9	1
NAM	all target files	252	3
OC1-OC15	BASE1.DAT	153,158,163,168 173,178,183,188, 193,198,203,208, 213,218,223	4
OD	RAD.DAT	179	1
OL	POL.DAT	189	5
ON	POL.DAT	185	3
OR	ROC.DAT	179	5
OT1-OT15	BASE1.DAT	152,157,162,167,172, 177,182,187,192,197, 202,207,212,217,222	1
OV	POL.DAT	180	5
OY	POL.DAT	179	1
O5	POL.DAT	188	1
PA1	BASE1.DAT	92	4
PA2	BASE1.DAT	106	4
PC1	BASE1.DA	84	4
PC2	BASE1.DAT	98	4
PR1	BASE1.DA	83	1
PR2	BASE1.DAT	97	1

Field Name	File Name	Starting Byte	Length
PT1	BASE1.DAT	82	1
PT2	BASE1.DAT	96	1
PX1	BASE1.DAT	88	4
PX2	BASE1.DAT	102	4
RA	BASE1.DAT	17	9
RC	BASE1.DAT	45	1
RI	BASE1.DAT	1	4
RN	BASE1.DAT	5	12
RO	BASE1.DAT	26	9
RP	BASE1.DAT	52	4
RS	BASE1.DAT	56	1
RT	BASE1.DAT	46	6
RU	BASE1.DAT	35	10
RY	BASE1.DAT	227	3
RZ	BASE1.DAT	230	3
SR	SAM.DAT	180	5
ST	SAM.DAT	179	1
TB	all target files	1	10
TD	TRK.DAT	179	5
TE	all target files	24	1
TF	TNK.DAT	179	1
TG	all target files	44	9
TJ	TNK.DAT	180	3
TL	all target files	35	9
TM	all target files	63	3
TN	all target files	11	12
TP	all target files	23	1
TR	all target files	27	1
TT	all target files	28	7
TTC	all target files	251	1
TU	all target files	53	10
TW	all target files	26	1
T3	all target files	176	3
UA	UNIT.DAT	7	10
UA1	UNIT.DAT	177	1
UA2	UNIT.DAT	208	1
UB	UNIT.DAT	17	4
UC	UNIT.DAT	1	6
UF1	UNIT.DAT	183	3
UF2	UNIT.DAT	214	3
UG1	UNIT.DAT	198	9
UG2	UNIT.DA	229	9
UH1	UNIT.DAT	28	72
UH2	UNIT.DAT	104	72

Field Name	File Name	Starting Byte	Length
UL1	UNIT.DAT	179	1
UL2	UNIT.DAT	210	1
UN1	UNIT.DAT	180	3
UN2	UNIT.DAT	211	3
UO1	UNIT.DAT	178	1
UO2	UNIT.DAT	209	1
UP	UNIT.DAT	21	3
UR1	UNIT.DAT	25	3
UR2	UNIT.DAT	101	3
UT1	UNIT.DAT	24	1
UT2	UNIT.DAT	100	1
UU1	UNIT.DAT	189	9
UU2	UNIT.DAT	220	9
UY1	UNIT.DAT	176	1
UY2	UNIT.DAT	207	1
UZ1	UNIT.DAT	186	3
UZ2	UNIT.DAT	217	3
WC1	BASE1.DAT	118	1
WC2	BASE1.DAT	132	1
WC3	BASE1.DAT	146	L
WO1	BASE1.DAT	120	3
WO2	BASE1.DAT	134	3
WO3	BASE1.DAT	148	3
WP1	BASE1.DAT	110	5
WP2	BASE1.DAT	124	5
WP3	BASE1.DAT	138	5
WT1	BASE1.DAT	123	1
WT2	BASE1.DAT	137	1
WT3	BASE1.DAT	151	1
WW1	BASE1.DAT	115	3
WW2	BASE1.DAT	129	3
WW3	BASE1.DAT	143	3

RECORD LAYOUTS

Filename: All Target Files
Record Length: 256 bytes
Organization: Sequential
Access: Sequential, Direct

Byte	Name	Attribute	Length	Format	Purpose
1-10	TB	byte	10	10A1	basic encyclopedia number
11-22	TN	byte	12	12A1	name
23	TP	byte	1	1A1	nomination indicator
24	TE	byte	1	1A1	enemy situation (order of battle)
25	TA	byte	1	1A1	assignment indicator
26	TW	byte	1	1A1	weather ceiling
27	TR	byte	1	1A1	priority
28-34	TT	byte	7	7A1	time last seen
35-43	TL	real	9	F9.4	latitude
44-52	TG	real	9	F9.4	longitude
53-62	TU	byte	10	10A1	UTM
63-65	TM	int	3	I3	mission pointer
66	AN1	byte	1	A1	aircraft type
67	A01	byte	1	1A1	standard configuration load code
68	AD1	byte	1	A1	dive angle
69	AF1	byte	1	A1	number of aircraft
70	AK1	byte	1	A1	knots true air speed
71	AI1	byte	1	A1	intevelometer
72-74	AA1	int	3	I3	altitude
75-76	AP1	int	2	I2	probability of damage
77	AN2	byte	1	A1	aircraft type
78	A02	byte	1	A1	SCL code
79	AD2	byte	1	A1	dive angle
80	AF2	byte	1	A1	number of aircraft
81	AK2	byte	1	A1	KTAS
82	AI2	byte	1	A1	intevelometer
83-85	AA2	int	3	I3	altitude
86-87	AP2	int	2	I2	probability of damage
88	AN3	byte	1	A1	aircraft type
89	A03	byte	1	A1	SCL code
90	AD3	byte	1	A1	dive angle
91	AF3	byte	1	A1	number of aircraft
92	AK3	byte	1	A1	KTAS
93	AI3	byte	1	A1	intevelometer
94-96	AA3	int	3	I3	altitude
97-98	AP3	int	2	I2	probability of damage

Byte	Name	Attribute	Length	Format	Purpose
99	AN4	byte	1	A1	aircraft type
100	AO4	byte	1	A1	SCL code
101	AD4	byte	1	A1	dive angle
102	AF4	byte	1	A1	number of aircraft
103	AK4	byte	1	A1	KTAS
104	AI4	byte	1	A1	intevelometer
105-107	AA4	int	3	I3	altitude
108-109	AP4	int	2	I2	probability of damage
110	AN5	byte	1	A1	aircraft type
111	AO5	byte	1	A1	SCL code
112	AD5	byte	1	A1	dive angle
113	AF5	byte	1	A1	number of aircraft
114	AK5	byte	1	A1	KTAS
115	AI5	byte	1	A1	intevelometer
116-118	AA5	int	3	I3	altitude
119-120	AP5	int	2	I2	probability of damage
121	AN6	byte	1	A1	aircraft type
122	AO6	byte	1	A1	SCL code
123	AD6	byte	1	A1	dive angle
124	AF6	byte	1	A1	number of aircraft
125	AK6	byte	1	A1	KTAS
126	AI6	byte	1	A1	intevelometer
127-129	AA6	int	3	I3	altitude
130-131	AP6	int	2	I2	probability of damage
132	AN7	byte	1	A1	aircraft type
133	AO7	byte	1	A1	SCL code
134	AD7	byte	1	A1	dive angle
135	AF7	byte	1	A1	number of aircraft
136	AK7	byte	1	A1	KTAS
137	AI7	byte	1	A1	intevelometer
138-140	AA7	int	3	I3	altitude
141-142	AP7	int	2	I2	probability of damage
143	AN8	byte	1	A1	aircraft type
144	AO8	byte	1	A1	SCL code
145	AD8	byte	1	A1	dive angle
146	AF8	byte	1	A1	number of aircraft
147	AK8	byte	1	A1	KTAS
148	AI8	byte	1	A1	intevelometer
149-151	AA8	int	3	I3	altitude
152-153	AP8	int	2	I2	probability of damage
154	AN9	byte	1	A1	aircraft type
155	AO9	byte	1	A1	SCL code
156	AD9	byte	1	A1	dive angle
157	AF9	byte	1	A1	number of aircraft
158	AK9	byte	1	A1	KTAS
159	AI9	byte	1	A1	intevelometer

Byte	Name	Attribute	Length	Format	Purpose
160-162	AA9	int	3	I3	altitude
163-164	AP9	int	2	I2	probability of damage
165	AN10	byte	1	A1	aircraft type
166	AO10	byte	1	A1	SCL code
167	AD10	byte	1	A1	dive angle
168	AF10	byte	1	A1	number of aircraft
169	AK10	byte	1	A1	KTAS
170	AI10	byte	1	A1	intevelometer
171-173	AA10	int	3	I3	altitude
174-175	AP10	int	2	I2	probability of damage
176-178	T3	int	3	J3	graphics id

ART.DAT

179	AV	byte	1	A1	artillary sub-type
180-250	GARB	byte	71	71A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	1A1	"000
256	B	byte	1	1A1	"013

EW.DAT

179	ET	byte	1	A1	early warning
180	ER	byte	1	A1	radar type
181-250	GARB	byte	70	70A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013

IND.DAT

179	IT	byte	1	A1	type of industrial target
180	IC	byte	1	A1	construction
181-183	IL	int	3	I3	length
184-186	IW	int	3	I3	width
187-250	GARB	byte	64	64A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013

Byte	Name	Attribute	Length	Format	Purpose
MIL.DAT					
179	MJ	byte	1	1A1	type of military target
180-184	MP	int	5	I5	length
185-189	MW	int	5	I5	width
190-250	GARB	byte	61	61A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013
POL.DAT					
179	OY	byte	1	1Ak1	type of POL target
180-184	'OV	int	5	I5	volume
185-187	ON	int	3	I3	number
188	O5	byte	1	A1	construction
189-193	OL	int	5	I5	length
194-250	GARB	byte	57	57A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013
RAD.DAT					
179	OD	byte	1	1A1	type of radar
180-250	GARB	byte	71	71A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013
ROC.DAT					
179-183	OR	int	5	I5	number of rockets
184-250	GARB	61	61A1	blanks	
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013

Byte	Name	Attribute	Length	Format	Purpose
SAM.DAT					
179	ST	byte	1	A1	type of sam
180-184	SR	int	5	I5	radar
185-250	GARB	byte	66	66A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013
TNK.DAT					
179	TF	byte	1	A1	type of tank
180-182	TJ	in	3	I3	number of tanks
183-250	GARB	byte	68	68A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013
AAA.DAT					
179	AG	byte	1	A1	type of AAA
180-250	GARB	byte	71	71A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013
BRI.DAT					
179	BT	byte	1	A1	type of bridge
180	BC	BYTE	1	A1	CONSTRUCTION
181-185	BL	int	5	I5	length
186-190	BW	int	5	I5	width
191-250	GARB	byte	60	60A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013

Byte	Name	Attribute	Length	Format	Purpose
TRK.DAT					
179-183	TD	int	5	I5	number of trucks
184-250	GARB	byte	67	67A1	blanks
251	TTC	byte	1	1A1	generic type code
252-254	NAM	byte	3	3A1	generic type
255	A	byte	1	A1	"000
256	B	byte	1	A1	"013

Filename: BASE1.DAT
Record Length: 256
Organization: Indexed
Access: Keyed
Primary Key: RI

Byte	Name	Attribute	Length	Format	Purpose
1-4	RI	byte	4	4A1	ICAO name
5-16	RN	byte	12	12A1	airfield name
17-25	RA	real	9	F9.4	latitude
26-34	RO	real	9	F9.4	longitude
35-44	RU	byte	10	10A1	UTM
45	RC	byte	1	1A1	fighter class
46-51	RT	byte	6	6A1	tower call sign
52-55	RP	byte	4	4A1	ramp capacity
56	RS	byte	1	1A1	status
57	AT1	byte	1	1A1	a/c type
58-61	AC1	integer	4	1I4	number of a/c
62	AT2	byte	1	1A1	a/c type
63-66	AC2	integer	4	1I4	number of a/c
67	AT3	byte	1	1A1	a/c type
68-71	AC3	integer	4	1I4	number of a/c
72	AT4	byte	1	1A1	a/c type
73-76	AC4	integer	4	1I4	number of a/c
77	AT5	byte	1	1A1	a/c type
78-81	AC5	integer	4	1I4	number of a/c
82	PT1	byte	1	1A1	type of POL
83	PR1	byte	1	1A1	resupply rate
84-87	PC1	int	4	1I4	capacity
88-91	PX1	int	4	1I4	current amount
92-95	PA1	int	4	1I4	resupply amount
96	PT2	byte	1	1A1	type of POL
97	PR2	byte	1	1A1	resupply rate
98-101	PC2	int	4	1I4	capacity
102-105	PX2	int	4	1I4	current amount
106-109	PA2	int	4	1I4	resupply amount
110-114	WP1	int	5	1I5	runway length
115-117	WW1	int	3	1I3	runway width
118	WC1	byte	1	1A1	surface cond
119	WS1	byte	1	1A1	surface
120-122	WO1	int	3	1I3	orientation
123	WT1	byte	1	1A1	status
124-128	WP2	int	5	1I5	runway length
129-131	WW2	int	3	1I3	runway width
132	WC2	byte	1	1A1	surface cond
133	WS2	byte	1	1A1	surface
134-136	WO2	int	3	1I3	orientation

Byte	Name	Attribute	Length	Format	Purpose
137	WT2	byte	1	1A1	status
138-142	WP3	int	5	1I5	runway length
143-145	WW3	int	3	1I3	runway width
146	WC3	byte	1	1A1	surface cond
147	WS3	byte	1	1A1	surface
148-150	WO3	int	3	1I3	orientation
151	WT3	byte	1	1A1	status
152	OT1	byte	1	1A1	ordnance type
153-156	OC1	int	4	1I4	# of ordnance
157	OT2	byte	1	1A1	ordnance type
158-161	OC2	int	4	1I4	# of ordnance
162	OT3	byte	1	1A1	ordnance type
163-166	OC3	int	4	1I4	# of ordnance
167	OT4	byte	1	1A1	ordnance type
168-171	OC4	int	4	1I4	# of ordnance
172	OT5	byte	1	1A1	ordnance type
173-176	OC5	int	4	1I4	# of ordnance
177	OT6	byte	1	1A1	ordnance type
178-181	OC6	int	4	1I4	# of ordnance
182	OT7	byte	1	1A1	ordnance type
183-186	OC7	int	4	1I4	# of ordnance
187	OT8	byte	1	1A1	ordnance type
188-191	OC8	int	4	1I4	# of ordnance
192	OT9	byte	1	1A1	ordnance type
193-196	OC9	int	4	1I4	# of ordnance
197	OT10	byte	1	1A1	ordnance type
198-201	OC10	int	4	1I4	# of ordnance
202	OT11	byte	1	1A1	ordnance type
203-206	OC11	int	4	1I4	# of ordnance
207	OT12	byte	1	1A1	ordnance type
208-211	OC12	int	4	1I4	# of ordnance
212	OT13	byte	1	1A1	ordnance type
213-216	OC13	int	4	1I4	# of ordnance
217	OT14	byte	1	1A1	ordnance type
218-221	OC14	int	4	1I4	# of ordnance
222	OT15	byte	1	1A1	ordnance type
223-226	OC15	int	4	1I4	# of ordnance
227-229	RY	int	3	I3	graphics id
230-232	RZ	int	3	1I3	last rep. day
233-235	PTR	int	3	1I3	unit pointer
236-237	CT	int	2	1I3	unit count
238-254	GARB	byte	17	17A1	padding
255	A	byte	1	1A1	"000
256	B	byte	1	1A1	"013

Filename: UNIT.DAT
Record Length: 256 bytes
Organization: Sequential
Access: Sequential, direct

Byte	Name	Attribute	Length	Format	Purpose
1-6	UC	byte	6	6A1	unit call sign
7-16	UA	byte	10	10A1	unit name
17-20	UB	byte	4	4A1	base ICAO
21-23	UP	int	3	1I3	total ac possessed by this unit
24	UT1	byte	1	1A1	operationally ready ac type
25-27	UR1	int	3	1I3	# ac of this type
28-99	UH1 (24)	int	72	24I3	# of ac in 24 hour time periods
100	UT2	byte	1	1A1	operationally ready ac type
101-103	UR2	int	3	1I3	# ac of this type
104-175	UH2 (24)	int	72	24I3	# of ac in 24 hour time periods
176	UY1	byte	1	1A1	alert ac type
177	UA1	byte	1	1A1	alert type (ground/air)
178	UO1	byte	1	1A1	ordnance type
179	UL1	byte	1	1A1	readiness indicator
180-182	UN1	int	3	1I3	# of ac possessed
183-185	UF1	int	3	1I3	time from target to base
186-188	UZ1	int	3	1I3	time to base to target
189-197	UU1	real	9	F9.4	latitude
198-206	UG1	real	9	F9.4	longitude
207	UY2	byte	1	1A1	alert ac type
208	UA2	byte	1	1A1	alert type (ground/air)
209	UO2	byte	1	1A1	ordnance type
210	UL2	byte	1	1A1	readiness indicator
211-213	UN2	int	3	1I3	# of ac possessed
214-216	UF2	int	3	1I3	time from target to base
217-219	UZ2	int	3	1I3	time to base to target
220-228	UU2	real	9	F9.4	latitude
229-237	UG2	real	9	F9.4	longitude
238-239	CN	int	2	J2	call sign #
240-254	GARB	byte	15	15A1	padding
255	A	byte	1	1A1	"000
256	B	byte	1	1A1	"013

Filename: MISION.DAT
Record Length: 256 bytes
Organization: Indexed
Access: Keyed
Primary Key: MN
Alternative Key 1: ME
Alternative Key 2: MF

Byte	Name	Attribute	Length	Format	Purpose
1-3	MN	byte	3	3A1	mission number
4-6	MY	byte	3	3A1	mission type
7-10	MK	int	4	I4	take-off time
11-14	MH	int	4	I4	time over target to
15-17	MA	int	3	I3	number of aircraft
18	MC	byte	1		1A1aircraft type
19	MO	byte	1	1A1	ordnance type
20-23	MI	byte	4	4A1	ICAO name
24-29	MS	byte	6	6A1	unit call sign
30-39	MB	byte	10	10A1	target BE number
40-42	MT	byte	3	3A1	target type
43-44	MD	int	2	I2	probability of damage
45-46	MQ	int	2	I2	optimal unit address
47-49	MR	int	3	I3	target graphics id
50-53	MG	int	4	I4	return time
54-57	MU	int	4	I4	time over target from
58	ME	byte	1	1A1	mission state
59-78	MN	byte	20	20A1	squad name
79-80	MV	int	2	I2	dive angle
81-82	MX	int	2	I2	intevelometer
83-86	ML	int	4	I4	altitude
87-89	MZ	int	3	I3	KTAS
90	MF	byte	1	1A1	fragged mission flag
91	M1	byte	1	1A1	display/remove flag (allocation pairing lines)
92-254	GARB	byte	165	165A1	blanks
255	A	byte	1	1A1	"000
256	B	byte	1	1A1	"013 end of record

Filename: DBFILE.DAT
Record Length: 256 bytes
Organization: Sequential
Access: Sequential, direct

Header Record

Byte	Name	Attribute	Length	Format	Purpose
1-3	NT	int	3	I3	number of target records
4-6	NM	int	3	I3	number of mission records
7-9	NA	int	3	I3	number of friendly airbase records
10-12	NU	int	3	I3	number of unit records
13-15	NW	int	3	I3	number of weapon choice records
16-18	NB	int	3	I3	number of airbase choice records
19-21	NSM	int	3	I3	number of suggested mission records
22-24	NN	int	3	I3	total records in DBFILE.DAT
25-254	GARB	byte	230	230A1	blanks
255	A	byte	1	1A1	"000
256	B	byte	1	1A1	"013 end of record

NT Record

See Target files

NM Record

See MISION.DAT

NA Record

See BASE1.DAT

NO Record

See UNIT.DAT